Local Deformation Modelling for Non-Rigid Structure from Motion

João Renato Kavamoto Fayad

Queen Mary, University of London

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

Queen Mary, University of London

2013

Abstract

Reconstructing the 3D geometry of scenes based on monocular image sequences is a long-standing problem in computer vision. *Structure from motion* (SfM) aims at a data-driven approach without requiring *a priori* models of the scene. When the scene is rigid, SfM is a well understood problem with solutions widely used in industry. However, if the scene is non-rigid, monocular reconstruction without additional information is an ill-posed problem and no satisfactory solution has yet been found.

Current non-rigid SfM (NRSfM) methods typically aim at modelling deformable motion globally. Additionally, most of these methods focus on cases where deformable motion is seen as small variations from a mean shape. In turn, these methods fail at reconstructing highly deformable objects such as a flag waving in the wind. Additionally, reconstructions typically consist of low detail, sparse point-cloud representation of objects.

In this thesis we aim at reconstructing highly deformable surfaces by modelling them locally. In line with a recent trend in NRSfM, we propose a piecewise approach which reconstructs local overlapping regions independently. These reconstructions are merged into a global object by imposing 3D consistency of the overlapping regions. We propose our own local model – the Quadratic Deformation model – and show how patch division and reconstruction can be formulated in a principled approach by alternating at minimizing a single geometric cost – the image re-projection error of the reconstruction. Moreover, we extend our approach to dense NRSfM, where reconstructions are preformed at the pixel level, improving the detail of state of the art reconstructions.

Finally we show how our principled approach can be used to perform simultaneous segmentation and reconstruction of articulated motion, recovering meaningful segments which provide a coarse 3D skeleton of the object.

Contents

Acknowledgements

This work was partially funded by Fundação para a Ciência e a Tecnologia (FCT) under Doctoral Grant SFRH/BD/70312/2010 and by the European Research Council under ERC Starting Grant agreement 204871-HUMANIS. I thank these institutions for supporting research, specially FCT for investing in young Portuguese researches and assuring we have access to the best possible working conditions.

I would like to thank Lourdes Agapito, my supervisor, for accepting me as her student and guiding me over the past four years. Often we hear stories about careless supervisors, but at no time did I feel I was on my own. She always found time to meet and discuss ideas, and did everything she could to ensure all her group had the best possible conditions to develop not only their projects, but also themselves as researchers. Even when the whole group was working long hours to meet a deadline she was committed, working late hours if needed and I really appreciate that.

I would also like to thank Chris Russell. Ever since he joined the group he has made a big impact on everyone's research. Chris and I quickly developed new ideas and he have not stopped working together since. He has been like a second supervisor, and I have learned a lot from him.

I thank everyone else that has been part of this group over the years: Marco Paladini, Jae-Hak Kim, Ravi Garg, Nikos Pitelis, Tassos Roussos and Sara Vicente. Thank you for making our lab an enjoyable place to work, and for all the support and discussions we had. I would like to thank Marco in special. For some time the group was just the two of us, and as the most senior student he was always ready to help me in

the struggle that can be the PhD life. I also thank all the people who have visited the lab this past few years and the other students and post-docs in the school for contributing to this great enironment. Special thanks to Francesco Setti and Parthipan Siva, who would always be up for some sunday roast during deadline time. I thank all the support and administrative staff from EECS, who one way or another help make our research possible.

I thank Alessio Del Bue for recommending me to follow his steps and be a PhD student of Lourdes. From the times of my Masters' degree to this day he has always been a great source of knowledge and support.

I thank Alem's football group, for helping me keep my sanity with our weekly football games. I thank all of my friends, back in Portugal and in London, for helping me take my mind off research on the weekends I was not in the lab. I thank my parents, and my brother, who were always there for me and without whom I could not have been where I am now. And I thank Sara, my girlfriend, for enduring countless hours of video call over the internet, often struggling with the connection. It is not easy for two PhD students like us to find time to maintain a long distance relationship, but we have made it through.

Related Publications

- João Fayad, Alessio Del Bue, Lourdes Agapito, Pedro M.Q. Aguiar **"Non-Rigid Structure from Motion using Quadratic Deformation Models"**, in the Proceedings of the 10^{th} British Machine Vision Conference, London, UK, 2009.

- João Fayad, Alessio Del Bue, Lourdes Agapito **"Piecewise Quadratic Reconstruction of Non-Rigid Surfaces from Monocular Sequences"**, in the Proceedings of the 11^{th} European Conference on Computer Vision, Crete, Greece, 2010.

- Chris Russell, João Fayad, Lourdes Agapito **"Energy Based Multiple Model Fitting for Non-Rigid Structure from Motion"**, in the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, Colorado, 2011.

- João Fayad, Chris Russell, Lourdes Agapito **"Automated Articulated Structure and 3D Shape Recovery from Point Correspondences"**, in the Proceedings of the 13^{th} International Conference on Computer Vision, Barcelona, Spain, 2011.

- Chris Russell, João Fayad, Lourdes Agapito **"Dense Non-Rigid Structure From Motion"**, to appear in the Proceedings of 3D Imaging, Modeling, Processing, Visualization and Transmission, Zurich, Switzerland, 2012.

Chapter 1

Introduction

The ability to recover a 3D description of our world from 2D images has always played a central role in computer vision research. From Marr's seminal description [65] of the visual task of determining 3D shape from images as a 3 step process: from a primal sketch to a 3D model via a 2.5D sketch; to Maliks's formulation [62] of the '3 R's' of vision as three interactive processes – Recognition, Reconstruction and Reorganization; the idea of recovering the third dimension, which is lost when an image is formed, has been key for understanding our world from images.

From the motion of vehicles and people in an urban scene, to natural outdoor scenes such as a group of trees waving in the wind, our world is essentially dynamic. Objects can move with various degrees of complexity, ranging from (approximately) rigid motion, such as cars driving down a road, to the very complex non-rigid motion of a flag waving in the wind or of the human body. It is precisely this case of complex non-rigid, deformable or articulated motion that motivates the work in this thesis. This problem is extremely challenging – in the presence of non-rigid motion the recovery of 3D geometry from a sequence of images is an inherently ill-posed problem since different dynamic 3D geometries can give rise to the same images. The problem becomes particularly challenging when no initial model or prior information is known about the observed scene.

Figure 1.1: Example of capturing motions with a MoCap system.

Computer vision algorithms are intrinsically linked to visual sensors used to perceive the world around us. Typically, computer vision systems rely on a human-like approach to the visual perception task, where a passive sensor (the camera) forms an image based on the visible light reflected from the objects. However, some approaches go beyond this conventional view and replace or augment cameras with sensors that work on different principles. Due to the challenging nature of the 3D reconstruction problem, some approaches have instead used alternative sensors with relative success. An example of such methods are Motion Capture systems (MoCap). A typical MoCap setup includes a set of 6 to 12 infra-red cameras observing a predefined capture volume. The objects to be reconstructed, which may be rigid or deformable, are placed in this volume and infra-red reflective markers attached to their surfaces. Given the high number of synchronized cameras viewing the scene it is possible to recover the 3D coordinates of the markers in every frame by triangulation, given the 2D coordinates of the markers in the infra-red images as input (see Figure 1.1). These methods have been used in a wide range of fields such as the film industry, for animating computer

generated characters, or in biomechanical studies, for sports or medical analysis, to accurately measure the motion of subjects. Still, the requirement of a special setup and reflective markers greatly limits its applicability. For instance, in the study of athletic performance the requirement to wear reflective markers could result in limitations or changes in performance.

Recently, structured light cameras have enjoyed great success in particular due to the advent of Microsoft's Kinect low-cost sensor (see Figure 1.2). Kinect emits an infra-red pattern on the scene and observes, with an infra-red camera, the distortions of the pattern caused by the scene. Observation of the distortion allows the recovery of a depth value for every pixel in the infra-red camera. This result can be combined with a regular RGB camera to provide a full colour 3D reconstruction of the scene. The disadvantages of this system are that it has a relatively low resolution, it provides noisy output and was designed for small indoor environments, and is therefore unable to cope with a more challenging setup where different illumination and sources of infra-red light, such as ambient day light, would need to be taken into consideration. However, its low cost makes it an attractive alternative to MoCap systems.

Figure 1.2: Left: RGB image. Right: Depth map corresponding to the RGB image on the left acquired with Kinect.

Other systems are more closely related to the human visual system and rely on a stereo pair of RGB cameras to infer the depth of the observed pixels [84]. These systems acquire two images of a scene from slightly different view points, and trian-

gulate the 3D coordinates of the observed pixels to recover their depth. The biggest challenge is not so much in the estimation of the 3D position of a pair of pixels but in establishing the pixel correspondence between the two observed images – the so called *stereo-matching* problem.

While these systems with extra sensors have shown substantial success at coping with the challenges of 3D reconstruction, one of their biggest disadvantages is that, to this day, the most common setup for artificial vision, from television broadcast to mobile phones, is a single camera. For this reason, one of the most active areas of research in 3D reconstruction deals with the problem where the sequence of images is acquired by a single camera *i.e*. a *monocular* video sequence. In this setup, the most interesting case is when one knows neither the motion of the camera nor the scene to be reconstructed. The family of methods which provide a data driven approach to estimating both the camera motion and the 3D geometry of the scene is called *Structure from Motion* (SfM).

1.1 Structure from Motion

The most common framework in SfM assumes a single moving camera viewing a scene. The only input information are the 2D image coordinates of a set of points observed in the images. The aim is to simultaneously recover the 3D coordinates of the points while estimating the camera motion. Since recovering the 3D geometry of a generic non-rigid scene from a monocular sequence is an ill-posed problem, for many decades SfM approaches have focused on the more constrained problem when the camera observes a *rigid* scene.

If the camera is calibrated, *i.e*. its internal parameters are known, the reconstruction of a rigid scene is possible when two (or more) views of the scene are available [48]. When the camera is uncalibrated, its internal parameters must also be inferred based on the image sequence. Although reconstruction in this case was shown to be possi-

ble [60], and self-calibration methods have also been developed following the seminal work of Faugeras [33], one of the most influential works in the rigid SfM field was the factorization method of Tomasi and Kanade [93], which assumed an *orthographic* camera model. This camera model is a good approximation of the projective operation when the range of depths of the points to reconstruct is small when compared to their distance to the camera. The projection equation simplifies greatly since it becomes linear and there is no requirement for internal calibration. From this seminal work other factorization methods have followed, extending it to multiple independently moving objects [23], rigid objects linked by an articulation [106, 98], and also to the perspective camera case [88].

Rigid reconstruction from image sequences is now a well understood problem with several applications in industry. For instance, the commercial software Boujou [13] is routinely used by film makers in Hollywood as once the camera position and 3D geometry of the scene is known, it is possible to augment it with computer generated characters (see Figure 1.3). Other successful examples are large scale reconstruction projects, such as *Building Rome in a Day* [4]. These methods typically aim at reconstructing tourist landmarks such as the Coliseum in Rome or the Notre Dame cathedral in Paris, by processing a large database of pictures available in community photo collections on the internet (*e.g.* Flickr). Recent work in 3D reconstruction by Newcombe *et al.* [67] has shown how it is possible to acquire very detailed 3D reconstructions from monocular video in real time.

1.1.1 Non-Rigid Structure from Motion

Intuitively, rigid motion reconstruction from multiple views is possible because with every new image observed by the camera the knowledge of the underlying fixed 3D structure increases. On the other hand when dealing with non-rigid motion the underlying 3D structure is different every time an image is acquired. This makes the problem

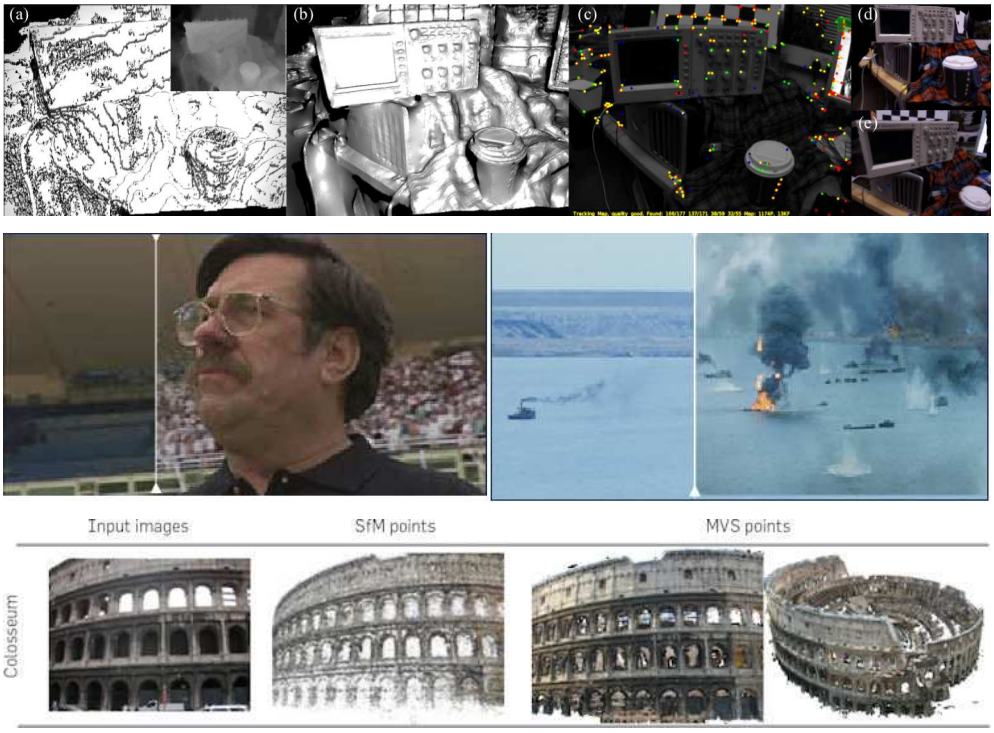


Figure 1.3: Top row: (a) Unregularized result from Newcombe *et al*. [67]; (b) Regularized result from [67], (c) Input video sequence;(d,e) Texture mapped reconstructions with [67]; Figure from Newcombe *et al*. [67]. Middle row: examples of film post-processing where computer generated objects are placed in the scene (right part of the image) vs the original footage (left part of the image). Image copyright 2d3 Ltd. [13]. Bottom row: Examples of an input image (left) and large scale reconstructions achieved with [4]. Figure from Agarwal *et al*. [4]

equivalent to 3D reconstruction from a single image, which, without any other prior information, is inherently ill-posed.

However objects do not change their shape randomly but instead deform according to their material properties and the laws of physics, which imposes constraints on the nature of their motion. This observation has been exploited to constrain the Non-Rigid Structure from Motion (NRSfM) problem by adding prior information to make the problem well posed.

The first successful approach to NRSfM was the seminal work by Bregler *et al*. [18]. In this work, the authors introduced a statistical prior by assuming the deformations of a non-rigid shape could be described by a low-rank shape basis model. Their assump-

tion was that non-rigid motion can be seen as small deviations from a mean shape, and the shape in each instant as a linear combination of K 3D shape bases. The low-rank shape basis model proved successful in the reconstruction of non-rigid sequences that fit into these assumptions, in particular face reconstruction. This model was very well received by the NRSfM community stemming several other works, which improved on [18] by proposing new optimisation strategies or additional model constraints [1, 104, 25, 96, 69, 7]. Other statistical priors proposed were a Gaussian distribution on the deformation coefficients of the model [95] , or a coarse-to-fine prior on the shape bases where each basis added should explain as much of the non-rigid motion variance as possible [18, 96, 7].

Another family of priors commonly used in NRSfM are the physical priors. These include temporal smoothness in the way the camera moves and the object deforms, and spatial smoothness of the object's surface [1, 26, 7, 96, 74, 34], inextensibility or local isometry constraints [100, 90], or assuming a a mixture of rigidly and non-rigidly moving points [26].

After a number of years when the low-rank shape basis model of Bregler *et al*. [18] has dominated the literature, it has become apparent that it can only model small linear deformations. Stronger linear or non-linear deformations would require a relatively large number of bases, violating the key low-rank assumption and leading to overfitting. Additionally the shape bases must be computed for every new sequence, and the reconstruction results are highly sensitive to the choice of K, which is difficult to estimate.

In response to these problems, other approaches have been recently proposed that depart from the low-rank shape basis model and target more complex non-rigid deformations [74, 6, 100, 90, 22, 36, 34]. A recent trend in NRSfM has been the emergence of *piecewise* approaches [100, 90, 22, 34]. The idea behind these methods is that in sequences where surfaces display very agile deformable motion with many strong local deformations, such as a flag waving in the wind, their high complexity makes

global modelling inherently ambiguous. This limitation applies to the rich body of work based on the low-rank shape basis model, as the complexity of such non-rigid motions would require a large number of shape bases which would rapidly lead to overfitting.

Piecewise methods split the points to be reconstructed into regions, each of which is modelled independently. Given a solution for each region, spatial consistency can be enforced between regions by requiring them to overlap and forcing 3D consistency between overlapping regions to create a continuous global surface. One exception is [22] where regions do not overlap and consistency is instead applied by forcing the regions to lie on a smooth surface. These methods differ mostly on the model chosen for the local regions, which can be rigid, planar [100, 22], locally triangular [89] or quadratic (see Chapters 4 to 6). However these methods suffer from a very important drawback as they overlook the problem of providing a principled formulation for the division of the surface into models. [100, 34] rely on a manual division of the surface, while in [90] the division comes directly from the choice of model and is formed by a Delaunay triangulation of the image correspondences. Only [22] provide a Markov Random Field (MRF) formulation where features are clustered into planar patches.

Following this recent trend of piecewise NRSfM methods, in this thesis we provide a unified principled formulation for this problem without assuming any division into local models *a priori*. Our formulation simultaneously divides the object into overlapping regions and reconstructs their 3D motion by minimizing the same geometric cost, the image reprojection error, in a hill-climbing approach.

We propose our own model for local regions, the Quadratic Deformation model, and provide experimental justification for our choice. While current NRSfM approaches are based on sparse point clouds, we show how our approach can be made computationally efficient to be used for dense NRSfM by reconstructing at pixel scale.

Finally we show how our principled piecewise approach is suited for simultaneous segmentation and 3D reconstruction of articulated motion. While with deformable sur-

faces the chosen regions represent sequence dependent local motion with no semantic meaning, in articulated motion the segmentation into links is important as it reveals the underlying 3D skeleton of the articulated object, which can also be automatically recovered.

1.2 Motivation

The single camera setup remains the most common and reliable form of acquiring images from a scene. In comparison with MoCap systems or the Kinect, a single camera setup is more portable, widely available, and works on a passive principle, meaning it is less invasive and has less influence on the scene we want to reconstruct. Additionally, as cameras have the same working principle as human vision, they can be used in any circumstances and environments that humans find themselves in. Furthermore, systems based on alternative sensors have the additional drawback that they can only deal with newly captured footage, and are unsuitable for the countless hours of archive footage from television broadcasts and films which display a great variety of subjects and scenes.

When examining the problem of rigid SfM, we realise that these methods have reached maturity and are now widely used in industry, which is in contrast with NRSfM methods. While rigid reconstruction can now be done in very large scale, or with great detail and even in real time, most non-rigid reconstruction methods are still only able to reconstruct a very sparse set of points, work mostly in batch approach and can only handle relatively small deformations. We take the success of rigid SfM as our motivation to bridge the gap between the rigid SfM and its non-rigid counterpart, as there is certainly a wealth of potential applications that could benefit from recovering the non-rigid shape from image sequences.

1.3 Applications

As mentioned before, rigid SfM methods have found its way into industry, with the most successful application being the inclusion of computer generated objects into previously acquired footage. To make the result realistic it is essential to accurately estimate the camera motion, otherwise the computer generated objects will not move in accordance with the original footage, or will require extensive manual intervention. Additionally, instead of building large and expensive sets it is now possible to build relatively smaller sets that focus on the action, and fill in the remaining scene with computer generated objects (see Figure 1.4).

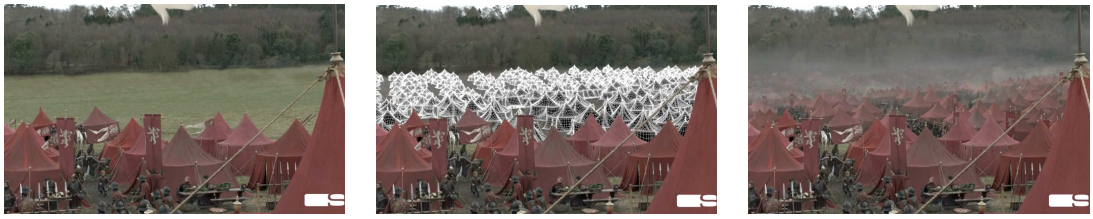


Figure 1.4: Example of how a small set can be augmented using computer generated objects. Image copyright HBO Entertainment and BlueBolt.

However, when it comes to non-rigid scenes such as the high detail deformations of the human face, state of the art methods rely on more complex and expensive setups. These setups can consist of: multiple synchronized cameras combined with special make-up to add texture to the subject [3]; hybrid methods that combine MoCap with synchronized cameras, resorting to active appearance models [54], or a previously acquired high density scan [9] to account for details; and coloured light photometric stereo [49].

The amount of research done in deformable surface reconstruction is a sign of the demand for these methods. While all these systems can provide good qualitative results (see Figure 1.5 top and middle) and have already been adopted by the film industry (see Figure 1.6 bottom), they have the aforementioned limitations of methods that require

Figure 1.5: Examples of dense face reconstruction methods. Top, from left to right: Input image with the tracked dots an texture; coarser large-scale reconstruction; reconstruction with added detail from the video model; realistic skin rendering; realistic skin rendering with different expression. Images from Bickel *et al*. [9]. Bottom, from left to right: dense maker placement for MoCap; motion transfer to animated characters. Images from Kholgade *et al*. [54].

additional sensors and setups. NRSfM methods can thus increase the applicability of such approaches by being less restrictive on the capture process, requiring only a single camera, resulting in a low-cost and less time consuming solution to this problem.

Reconstructing human deformable and articulated motion in high detail has also applications in the sports and health domains. In performance analysis or for the detection of pathologies, accurate motion reconstruction is very critical, which is why MoCap systems have been the main choice so far. As discussed before, MoCap systems require special setups and the need to attach reflective markers on the body of the subjects, preventing the analysis of motion in a more natural environment [108]. Ideally, it would be preferable to be able to analyse athletic performance during competition to have access to more meaningful data. While this is not possible with a MoCap system, it could be achieved if accurate deformable and articulated motion could be recovered from video footage of sports events.

Figure 1.6: Top: Make-up based motion capture where appearance model is learned. Middle: Stereo setup to record actor's performance. Bottom, from left to write: Body actor; Motion transfer to computer generated head; Final result with rendered detail. Image copyright Digital Domain

In addition, these 3D reconstruction techniques have also recently been applied to enhance visualisation in medical keyhole surgery. During these interventions, it is often helpful to be able to perform a 3D mapping of the target area. Since our body is made of soft tissues that undergo strong deformations, it is then crucial that these methods can recover the 3D geometry of deformable objects (see Figure 1.7).

1.4 Contributions

The aim of this work is to bridge the gap between rigid and non-rigid SfM. Typically NRSfM focuses on deformations that can be explained by the low-rank shape basis model, which means they must be small deviations from a mean shape. Additionally, this model is usually sensitive to the number of shape basis used, typically relying on

Figure 1.7: Reconstruction of an uterus from images acquired during a medical intervention. Left: Feature tracks over the sequence. Centre: Rigid reconstruction of the object. Right: Surface parametrization (top) and example of a non-rigid deformation the uterus can undergo (bottom).Figure from Bartoli *et al*. [8]

the user to specify it. In line with a recent trend in NRSfM [100, 90, 34, 22], we argue that non-rigid motion is best modelled *locally*. In contrast with other piecewise approaches, we formulate our piecewise NRSfM problem as a labelling problem, providing a principled formulation for the simultaneous 3D reconstruction and division into patches. We now summarise our contributions to the NRSfM problem:

- In Chapter 3 we introduce the Quadratic Deformation (QD) model, a physically grounded deformation model, into the NRSfM formulation. We show how this allows the 3D reconstruction of non-linear deformations viewed by an orthographic camera. Unlike the low-rank shape basis model, the QD model is of fixed rank, and so there is no need for the user to specify any parameters. We formulate the NRSfM problem using a non-linear optimisation scheme to minimize image reprojection error and recover the 3D geometry of the object [36].

- In Chapter 4 we argue that local modelling of non-rigid motion leads to more accurate reconstructions than global modelling in sequences with strong deformations and agile motions. We propose a piecewise NRSfM formulation which divides the surface into overlapping patches, reconstructs each of them individually, and finally merges all the patches imposing the constraint that points shared

by patches must correspond to the same 3D points in space. Our method is generic in the sense that it does not rely on any specific reconstruction method for the individual patches. While any SfM method could be used to reconstruct those patches, we support our choice of the QD model with experiments. We show how the independent patches can be stitched back together and propose a final optimisation step to refine the surface reconstruction jointly [34].

- In Chapter 5 we propose an energy-based geometric multiple model fitting formulation to the piecewise NRSfM problem as a principled method to divide the object into regions fit for local modelling. Inspired by recent advances in multiple model fitting [53], we formulate the NRSfM problem as a labelling problem where both the labels (model parameters) and their assignment to data points are computed simultaneously. A fundamental requirement of our piecewise reconstruction is the need for overlap between models to enforce global consistency, and to encourage smooth transitions between models. We capture this in our formulation by changing the classical labelling paradigm and allowing feature points lying in the boundary between models to have more than one label or, equivalently, to belong to more than one model [77].

- In Chapter 6 we show how our multiple model fitting approach can be extended to template-free dense reconstruction by providing asymptotic improvements to current optimisation approaches, allowing our method to scale to the dense case. We tackle the limitations of computing a rest shape based on available rigid motion, making the method applicable in more general sequences and increasing its robustness [78].

- In Chapter 7 we show how our energy-based geometric multiple model fitting with overlapping models can also be used to perform simultaneous segmentation and 3D reconstruction of articulated motion. By treating an articulated object as a set of rigid links, we show how fitting rigid models to the data provides a

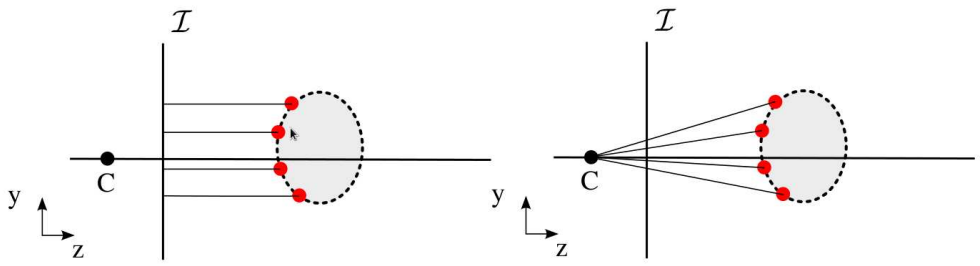
segmentation of the object where the overlapping regions naturally become the articulations, allowing to automatically recover a 3D skeleton of the articulated object [37].

Chapter 2

Literature Review

As discussed in Chapter 1, recovering the 3D geometry of an observed scene is a key problem in computer vision. When considering a generic scene which can be composed of multiple objects with a wide range of motion complexities, recovering the 3D geometry from images without any additional assumptions is an inherently ill-posed problem, as there are many 3D geometries that can give origin to the same images. To tackle these limitations, some methods propose the use of alternative sensors (*i.e.* not regular RGB cameras) such as MoCap systems or range cameras like Microsoft's Kinect (see Chapter 1).

Still, there are other families of methods using different constraints to recover the 3D geometry of a scene. One of such approaches is, for instance, the family of methods known as *photometric stereo* [103]. These methods recover surface normals of an object by establishing a relationship between the light reflected by the object and the surface normal. By acquiring a set of (at least 3) images from a given viewpoint while changing the lighting conditions, these methods are able to recover the 3D geometry of *static scenes*, as the geometry must be constant while the lighting changes. However it can be modified for the non-rigid motion case by illuminating a non-rigid object with three light sources of different colour (usually red, green and blue) which shine from different positions simultaneously [49]. Other methods such as *shape from shading*



(a) Orthographic projection (b) Perspective projection

Figure 2.1: (a) Orthographic projection assumes rays from the object are parallel to the image plane $\mathcal{I}$. (b) Perspective projection considers the rays from the object to converge at the camera centre **C**. While cameras are mostly perspective, orthographic projection is a good approximation when the relative depths of the object are small compared to the distance to the camera.

explore a similar relationship between the reflected light and the surface normal of objects when when the position and intensity of the light source is known [51].

However, the focus of this thesis is on the 3D reconstruction of non-rigid motion when viewed by a single moving camera – a problem known as *Non-Rigid Structure from Motion* (NRSfM).

2.1 Factorization for Rigid SfM

Before we go into the non-rigid reconstruction literature, we will describe the seminal work of Tomasi and Kanade [93] as its framework is common to many NRSfM methods, including the low-rank shape basis model of Bregler *et al*. [18] and the QD model we will present in Chapter 3. In the Tomasi and Kanade [93] framework a group of P points belonging to a rigid object is observed over F images by an *orthographic* camera (see Figure 2.1).

In the orthographic camera case, the projection matrix is defined as:

$$\Pi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{2.1}$$

which essentially truncates the contribution to the projection operation played by the third coordinate of the 3D point. A rigid object composed of P feature points is represented by a $3 \times P$ matrix

$$\mathtt{S}_r = \begin{bmatrix} X_1 & X_2 & \dots & X_P \\ Y_1 & Y_2 & \dots & Y_P \\ Z_1 & Z_2 & \dots & Z_P \end{bmatrix}, \tag{2.2}$$

where the 3D coordinates of the points in a given local referential are stacked column-wise. Rigid body motion is described by a rotation and translation. Thus the orthographic projection of a 3D point can be described, at every image i, by

$$\mathtt{W}_i = \Pi \mathtt{R}_i \mathtt{S}_r + \mathtt{T}_i, \tag{2.3}$$

where $\mathtt{R}_i$ describes the 3×3 relative rotation (*i.e.* $\mathtt{R}_i\mathtt{R}_i^T = \mathtt{R}_i^T\mathtt{R}_i = \mathtt{I}_{3\times3}$) between the rigid object and the camera, and where $\mathtt{T}_i$ is a $2 \times P$ matrix where every column is equal to $\mathbf{t}_i$. Each vector $\mathbf{t}_i$ can be computed as the centroid of the point cloud $\mathtt{W}_i$ and thus can be easily eliminated by subtracting the coordinates of the centroid of the point cloud. Therefore, instead of considering $\mathtt{W}_i$, a registered form of this matrix is used:$\tilde{\mathtt{W}}_i = \mathtt{W}_i - \mathtt{T}_i$. Stacking these equations over all i frames results in:

$$\tilde{\mathtt{W}} = \begin{bmatrix} \tilde{\mathtt{W}}_1 \\ \tilde{\mathtt{W}}_2 \\ \vdots \\ \tilde{\mathtt{W}}_F \end{bmatrix} = \begin{bmatrix} \Pi\mathtt{R}_1 \\ \Pi\mathtt{R}_2 \\ \vdots \\ \Pi\mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \cdots & \mathbf{s}_P \end{bmatrix} = \mathtt{MS}, \tag{2.4}$$

From Equation 2.4 one can see that if $F \gg 3$ and $P \gg 3$, which is true for interesting sequences, then $rank(\tilde{\mathtt{W}}) \leq 3$ (and $rank(\mathtt{W}) \leq 4$). These rank properties constitute the basis of the rigid factorization algorithm [93]. Applying the rank constraint, the $rank-3$ truncated SVD decomposition of $\tilde{\mathtt{W}}$ to factorize it into a product of two terms:

$$\tilde{\mathtt{W}} \approx \mathtt{U}_3 \, \Sigma_3 \, \mathtt{V}_3^T = \mathtt{U}_3 \, \Sigma_3^{1/2} \Sigma_3^{1/2} \, \mathtt{V}_3^T = \hat{\mathtt{M}} \, \hat{\mathtt{S}}, \tag{2.5}$$

where $\mathtt{U}_3$ is a $3F \times 3$ matrix, $\mathtt{V}_3$ a $P \times 3$ matrix and Σ_3 is a 3×3 diagonal matrix, all resulting from the truncated SVD decomposition of $\tilde{\mathtt{W}}$. The matrices $\hat{\mathtt{M}}$ and $\hat{\mathtt{S}}$ are the affine versions of $\mathtt{M}$ and $\mathtt{S}$, *i.e.* they do not preserve the angles and lengths of the original 3D object, as there is an inherent ambiguity in this factorization:

$$\hat{\mathtt{M}} \, \hat{\mathtt{S}} = \hat{\mathtt{M}} \, \mathtt{H} \, \mathtt{H}^{-1} \, \hat{\mathtt{S}} = \mathtt{M} \, \mathtt{S}, \tag{2.6}$$

where $\mathtt{H}$ is any 3×3 invertible matrix. Equation 2.4 shows that the desired solution for the motion matrix $\mathtt{M}$ has a very specific structure, which preserves the metric properties. The ambiguity is then resolved by finding the transformation $\mathtt{H}$ that will bring $\hat{\mathtt{M}}$ into its correct structure – a step commonly referred to as the *metric upgrade*. This can be done by enforcing the following metric constraints for every frame i:

$$\hat{\mathbf{m}}_{ik}^T \, \mathtt{H}\mathtt{H}^T \, \hat{\mathbf{m}}_{ik} = 1, \tag{2.7}$$

$$\hat{\mathbf{m}}_{ik}^T \, \mathtt{H}\mathtt{H}^T \, \hat{\mathbf{m}}_{il} = 0, \; l \neq k, \tag{2.8}$$

where $\hat{\mathbf{m}}_{ik}$ and $\hat{\mathbf{m}}_{il}$ are respectively the k-th and l-th row of matrix $\hat{\mathtt{M}}_i$ $(k, l = 1, 2)$.

2.2 Non-Rigid Structure from Motion and the Low-Rank Shape Basis Model

Consider a similar framework to that of Tomasi and Kanade [93], where P points are tracked along F images, captured by an orthographic camera. However, we now consider a non-rigid shape that varies from frame to frame, *i.e*. in the non-rigid motion case one cannot think about a single 3D configuration of the object as this configuration is, in general, different for every frame i.

Based on Equation 2.3, we can then write the image coordinates of a non-rigid motion under orthography as

$$\mathtt{W}_i = \Pi \mathtt{R}_i \mathtt{S}_i + \mathtt{t}_i. \tag{2.9}$$

Equation 2.9 shows that in general we must recover both the time-evolving 3D shape and the relative camera motion matrices per frame. Therefore, for each frame i, we would have to estimate $3P$ shape coordinates, 4 independent parameters for the rotation matrix and 2 parameters for the translation, given only $2P$ equations. In other words, this problem is equivalent to reconstructing the 3D geometry from a single 2D image.

Without any other constraints, this problem is ill-posed. However, objects do not deform randomly and in fact their motions are constrained by the laws of physics. This implies that the motion of points on a non-rigid surface is bound to be correlated and not completely independent. Such dependencies can then be explored in order to include more constraints into the problem and make it well-posed.

The first successful approach to non-rigid structure from motion was the factorization approach of Bregler *et al*. [18]. This method constrained the NRSfM problem by postulating that the 3D configurations of a non-rigid object can be described as a linear combination of K shape bases. This method is commonly referred to as the low-rank shape basis model.

2.2.1 Low-Rank Shape Basis Model

In the low-rank shape basis model of Bregler *et al*. [18] the 3D configuration of a non-rigid object at each frame i can be represented by the linear combination

$$\mathtt{S}_i = \sum_{d=1}^{K} a_{id}\, \mathtt{B}_d, \tag{2.10}$$

where $\mathtt{B}_d$ is the $3 \times P$ shape basis matrix, and a_{id} is a scalar deformation weight for base d at frame i. Stacking these equations over all frames eventually leads to the trilinear product

$$\begin{aligned} \tilde{\mathtt{W}} &= \begin{bmatrix} \Pi\mathtt{R}_1 & & & \\ & \Pi\mathtt{R}_2 & & \\ & & \ddots & \\ & & & \Pi\mathtt{R}_F \end{bmatrix} \begin{bmatrix} a_{11} & \dots & a_{1K} \\ \vdots & \ddots & \vdots \\ a_{F1} & \dots & a_{FK} \end{bmatrix} \begin{bmatrix} \mathtt{B}_1 \\ \vdots \\ \mathtt{B}_K \end{bmatrix} \\ &= \check{\mathtt{M}}\, \mathtt{A}\, \mathtt{B} = \mathtt{D}\, \mathtt{B}, \end{aligned} \tag{2.11}$$

where $\tilde{\mathtt{W}}$ is a $2F \times P$ matrix containing the stack of registered 2D coordinates, $\check{\mathtt{M}}$ is a diagonal arrangement of the stack of orthographic projection matrices in $\mathtt{M}$, $\mathtt{A}$ is a $2F \times 3K$ matrix which contains the deformation weights a_{ij}, and $\mathtt{B}$ is a $3K \times P$ matrix containing a stack of the K shape bases. Instead of the $rank - 3$ system of the rigid motion case, this formulation results in a $rank - 3K$ system. Unlike the rigid case, the value of K is not known *a priori* as it depends on the degree of non-rigidity of the motion.

Assume K is known. In analogy with the rigid factorization of Tomasi and Kanade [93], the parameters of this model are first estimated by performing a $rank - 3K$ truncated SVD on the data matrix:

$$\tilde{\mathtt{W}} \approx \mathtt{U}_{3K}\, \Sigma_{3K}^{1/2}\, \Sigma_{3K}^{1/2}\, \mathtt{V}_{3K}^{T} = \hat{\mathtt{D}}\, \hat{\mathtt{B}}. \tag{2.12}$$

As in the rigid factorization case (see Equation 2.6), there is also an ambiguity for every $3K \times 3K$ invertible matrix $\mathtt{G}$ such that

$$\mathtt{W} = \mathtt{D}\mathtt{B} = \hat{\mathtt{D}}\mathtt{G}\mathtt{G}^{-1}\hat{\mathtt{B}}. \tag{2.13}$$

Defining $\hat{\mathtt{D}}_i = [a_{i1}\hat{\mathtt{M}}_i \dots a_{iK}\hat{\mathtt{M}}_i]$ as the affine version of $\mathtt{D}_i$ where i represents the the frame index, and $\hat{\mathtt{M}}_i$ is the affine version of $\mathtt{M}_i$. If $\hat{\mathtt{D}}_i$ is rearranged into

$$\bar{\mathtt{D}}_i = \begin{bmatrix} a_{i1}\hat{r}_{i1} & a_{i1}\hat{r}_{i2} & \dots & a_{i1}\hat{r}_{i6} \\ a_{i2}\hat{r}_{i1} & a_{i2}\hat{r}_{i2} & \dots & a_{i2}\hat{r}_{i6} \\ \vdots & \vdots & \ddots & \vdots \\ a_{iK}\hat{r}_{i1} & a_{iK}\hat{r}_{i2} & \dots & a_{iK}\hat{r}_{i6} \end{bmatrix} = [a_{i1}a_{i2}\dots a_{iK}] \begin{bmatrix} \hat{r}_{i1} \\ \hat{r}_{i2} \\ \vdots \\ \hat{r}_{i6} \end{bmatrix} \tag{2.14}$$

one can see that the $K \times 6$ matrix $\bar{\mathtt{D}}_i$ is $rank - 1$, and so the deformation coefficients $\mathtt{A}_i$ and the affine matrices $\hat{\mathtt{M}}_i$ can be recovered by a performing a $rank - 1$ truncated SVD on every sub-matrix $\bar{\mathtt{D}}_i$. Finally, the metric upgrade step is performed as in the rigid case by estimating a single 3×3 matrix $\mathtt{H}$. Once the transformation $\mathtt{H}$ is recovered, it is applied to every basis k as $\mathtt{B}_k = \mathtt{H}^{-1}\hat{\mathtt{B}}_k$, effectively making $\mathtt{G}$ a block diagonal matrix (see reconstruction results in Figure 2.2).

The low-rank linear shape model proposed by Bregler *et al.* was quickly adopted by the NRSfM community as it provided a useful formulation to describe non-rigid motion. However its formulation as a succession of truncated SVD's means that in the presence of noise there will be consistent loss of information, which in turn will limit the methods applicability to the case of small deformations. In addition, their metric upgrade step is only an approximation, as in general $\mathtt{G}$ will not be block diagonal. Finally, it also requires full tracks for all the P points. Nonetheless this representation of non-rigid motion was well accepted by the community and led to alternative approaches that tackled its disadvantages. We will classify these approaches according to their optimisation strategy into *closed-form methods*, *alternation methods* and

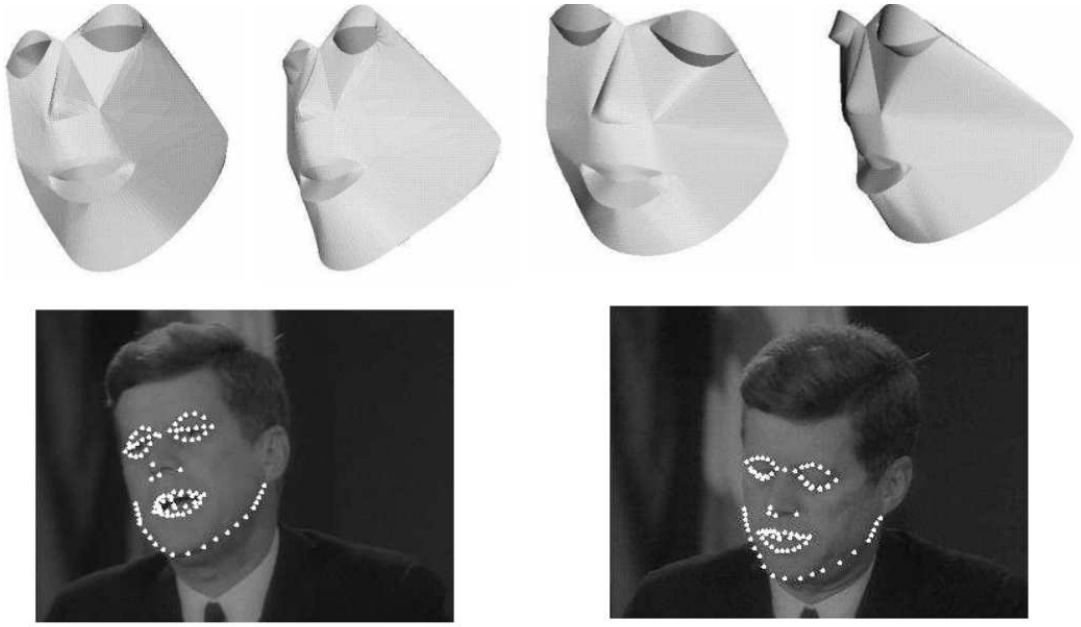


Figure 2.2: Results of a face reconstruction form Bregler *et al*. [18]. Figure from Bregler *et al*. [18].

non-linear least-squares methods.

Closed-form methods

With the success of the non-rigid factorization algorithm of Bregler *et al*. [18], other approaches emerged that proposed closed-form solutions to the factorization problem. These approaches mainly focus on how to estimate the $3K \times 3K$ metric upgrade matrix `G` explicitly (see Equation 2.13).

Xiao *et al*. [104] studied the properties of `G` and the metric constraints and argued that the orthonormality constraints were insufficient in the case of non-rigid factorization as the solution to this equation was ambiguous, containing valid an invalid sets of basis. To ensure the selection of a valid set of basis, Xiao *et al*. [104] observed that, under the low-rank shape basis assumption, the set of possible deformable shapes lie in a K-bases linear space. Therefore, any K independent bases from that space form a valid bases set. They thus propose to determine the K basis from the data, requiring

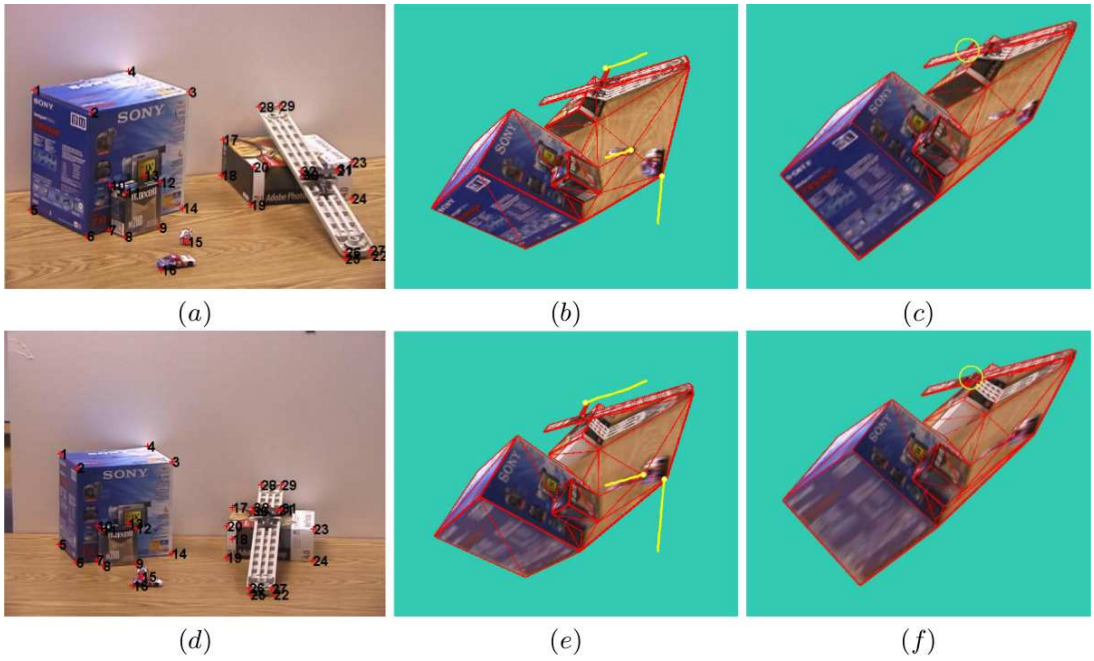


Figure 2.3: Results form Xiao *et al*. [104] on three independently moving objects. (a,d) Two images of the scene. (c,d) Reconstructions from Xiao *et al*. [104]. (e,f) Reconstructions from Brand [16]. Figure from Xiao *et al*. [104].

that each one of the K basis is independently observed in at least one image. The independent shapes are detected by finding the $2K \times P$ sub-matrix of $\tilde{\mathtt{W}}$ which results in the lowest condition number.

[104] showed exact results in synthetic data when their basis constraints could be applied. However this method turned out to be very sensitive to the choice of the K basis set. Additionally, it assumes close to perfect tracking, being its performance severely affected by noise, and it being unable to handle outliers or missing tracks.

In an attempt to solve the problems with the basis constraint and the sensitivity to noise, Brand [17] later proposed a different closed-form solution to the NRSfM problem. In [17], the metric upgrade is computed by estimating $\mathtt{G}\mathtt{G}^T$ via a formulation that minimizes the deviation of the current motion matrices from orthogonality. This results in an exact solution for noiseless data, although no such guarantee can be given for noisy conditions. Brand's method [17] showed better performance than [104] without the reliance on basis constraints. This was the first work to hint that orthonormality

constraints were in fact sufficient to impose the metric upgrade. However it still relies on close to perfect tracks.

It was not until 2009 that Akther *et al*. [5] in their work "In defence of orthonormality constraints for non-rigid structure from motion" showed that Xiao *et al*.'s formulation was incomplete as it failed to impose $rank - 3$ constraint on the metric upgrade matrix. The addition of this new result to the orthonormality constraints was shown to be sufficient to remove the ambiguity in the reconstructed 3D shape . However these constraints are non-linear and very difficult to optimise, which prevented the authors from proposing a closed-form solution. Instead they rely on non-linear optimisation schemes (see Section 2.2.1).

In the case where perspective effects cannot be ignored, Hartley and Vidal [46] proposed a closed-form linear solution. This algorithm requires the initial estimation of a multifocal tensor, for which a linear method exists [47]. The tensor is then factorized into the projection matrices and then simple linear algebraic techniques are used to enforce constraints on the projection matrices and estimate explicitly the corrective transformation. Although the entire approach is linear, the authors report that the initial tensor estimation and factorization is very sensitive to noise.

Closed-form solutions have the advantage of providing exact solutions to the NRSfM problem in the noiseless case and having, in general, a low computational cost. However these approaches tend to be very sensitive to noise. Other important limitations include the need for full tracks and the sensitivity to outliers. While these limitations can make them unappealing for real world applications, closed-form approaches are in general a good initialisation for other optimisation schemes and robust formulations (see Section 2.2.1).

Alternation methods

Alternation approaches solve the trilinear system of the low-rank shape basis model by estimating one of the factors (the camera motion matrices `M`, the linear deformation coefficients `A` or the shape bases `B`), while keeping the remaining two factors fixed. An alternation of the factor to estimate is performed until convergence.

The first of these approaches was proposed by Torresani *et al*. [97] who initialized the camera matrices `M` using the rigid factorization algorithm of Tomasi and Kanade [93]. The deformation coefficients in `A` were initialized to small random values, and thus `B` could be estimated by least-squares. While `A` and `B` are estimated in closed-form, that was not done for `M` due to the non-linear constraints of orthonormality. Instead, the camera matrices were parametrized with exponential map coordinates, and their update computed with a single Gauss-Newton step.

While [97] improved on previous approaches by explicitly imposing the metric constraints with the exponential map parametrisation of rotation matrices, their camera update step requires good initial estimates as it is only able to perform small changes to the initial value. Since the rotation matrices are initialized using the rigid factorisation method of Tomasi and Kanade [93], there is a strong assumption that the rigid motion of the object will be dominant as to provide a good estimate of those matrices.

Building on [97], Torresani *et al*. [96] proposed an approach where the low-rank basis shape model was replaced by a probabilistic PCA (PPCA) model, assuming a Gaussian distribution on the deformation weights. This Gaussian prior is an implicit assumption that the deformation weights will not vary much from the mean, resulting in small variations from the mean rigid shape. This formulation is solved using an EM algorithm, where the deformation weights become *latent variables* and are not explicitly solved for, resulting in less parameters to optimise.

The PPCA algorithm was also augmented with a Linear Dynamics model of the shape, where a temporal smoothness prior is added by parametrizing the Z coordinates

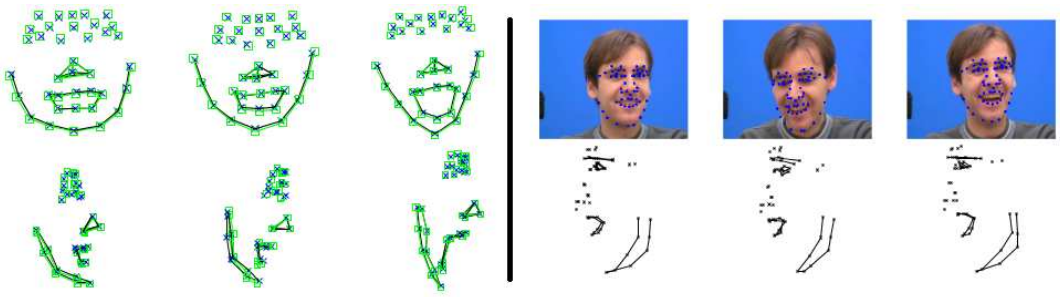


Figure 2.4: Results from Paladini *et al*. [69]. Left: Reconstruction of a MoCap sequence. Green squares show the ground truth positions while the blue crosses show the reconstruction results. Right: Reconstruction of a real sequence. Image from Paladini *et al*. [69].

of the points at frame i as a linear function of the coordinates at frame $i-1$. Since these methods had state of the art performance and resilience to missing tracks, after the authors released its source code it quickly became a benchmark for NRSfM. However, as [97], their method relies on the assumption of a strong rigid component as they model explicitly for a rigid mean shape.

Paladini *et al*. [69] proposed an alternation approach where the focus was in effectively recovering the metric structure of the motion matrix `M` (see Figure 2.4 for results). The authors showed that given an estimate for `D` and `B` (see Equation 2.12) solving for the camera matrices `M` with the orthonormality constraints is a non-convex problem. A tight relaxation of this problem was proposed that results in a semi-definite program (SDP) which is solved using SeDuMi [87]. Refinements of the estimates are computed by alternation, with the additional metric upgrade step where the camera matrices are projected onto the (Stiefel) manifold of orthonormal matrices. Del Bue *et al*. [27] later generalized [69] to any bilinear factorization approach under special manifold constraints.

Alternation methods provided important contributions to the NRSfM scene as they improved upon closed-form solutions. These algorithms provided the ability to handle missing data and higher resilience to noise on the tracks. Additionally, these approaches maintain the goal of computing the metric upgrade matrix `G` explicitly, which

the closed-form methods showed to improve the quality of the reconstructions. While some of the alternation steps can be solved linear, solving for orthonormal camera matrices rely on approximations that result in loss of accuracy and higher convergence time. In fact, the possibility of low convergence rate due to inadequate initialization or ambiguities in the motion are one of the major drawbacks of these solutions.

Non-linear least-squares methods

The non-linear optimisation approaches to NRSfM aim at simultaneously recovering all the parameters of the trilinear problem (`M`, `A`, `B`) in a single optimisation. In this framework, the NRSfM problem is formulated as the minimization of a geometric cost, the *re-projection error*, which measures the sum of squared differences between the measured image coordinates and the re-projection of the estimated 3D points onto the image

$$\sum_{i=1}^{F}\sum_{j=1}^{P} \|\tilde{\mathbf{w}}_{ij} - \Pi\mathbb{R}_i \sum_{d=1}^{K} a_{id}\ \mathbf{b}_{dj}\|^2. \tag{2.15}$$

The number of parameters to estimate in this formulation increases significantly as the sequences are longer and more points are tracked. This would make the reconstruction of interesting sequences impractical. However, most of these parameters do not interact with each other, as the motion parameters are image dependent. This results in a sparsity of the system, which is exploited by the *Bundle Adjustment* [99] non-linear optimisation algorithm to efficiently solve the problem of simultaneous motion and shape parameter estimation.

Unlike analytical solutions, Bundle Adjustment [99] cannot guarantee convergence to the global minimum of the cost-function. However, it can be efficiently used to refine approximate solutions that have been estimated analytically, as these provide a good initialisation for the non-linear optimisation method. The greatest advantage of formulating the NRSfM in this way is that prior information on the problem can be easily integrated into the optimisation by including additional terms into the cost-

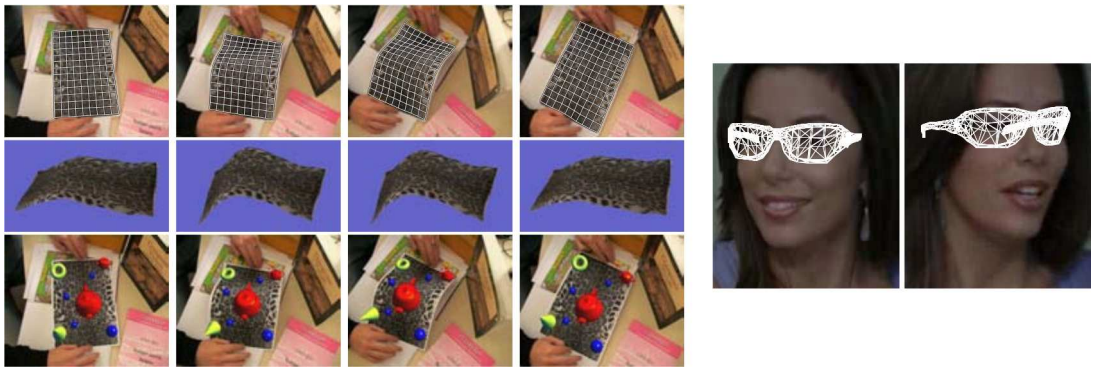


Figure 2.5: Reconstruction results from Bartoli *et al*. [7]. Left: reconstruction of a paper deforming sequence, with the re-projection of the reconstruction (top), 3D reconstruction (middle) and augmented reality example (bottom). Left: Augmented reality example by reprojecting a virtual object. Image from Bartoli *et al*. [7].

function. These costs need not be linear, and, although they increase the complexity of the system, in general the overall problem remains sparse and can be efficiently solved.

The first of these approaches was proposed by Aanæs and Khal [1], where they argued that temporal smoothness priors could be easily added to a Bundle Adjustment [99] formulation to regularise the reconstruction ambiguities of the NRSfM problem. Del Bue *et al*. [25], when focusing on face motion reconstruction, introduced a rigidity prior on the points lying on the head and nose, which was used to better disambiguate the rigid motion parameter estimation.

Bartoli *et al*. [7] proposed a different paradigm in the optimisation of the low-rank shape basis problem. Instead of simultaneously optimising for the parameters of the K shape basis, the authors proposed this optimisation to be done in a *coarse-to-fine* approach. In their formulation, shape basis are added and optimised incrementally so that every new basis explains as much as possible of the variance of the data that was not explained by previous basis. In addition, temporal and spatial smoothness terms were also added to the cost-function.

2.2.2 Alternative models for non-rigid structure from motion

Even though the low-rank shape basis model has been widely accepted and several other extensions and optimisation strategies for the problem have been proposed, to this day there is no generic solution to the NRSfM problem. As several authors argue [74, 45], this model is very sensitive to the choice of K. While an underestimation of K would result in poor reconstruction, since not enough deformation modes would be available, an overestimation of K would lead to overfitting, with the extra modes fitting to noise in the data. Additionally there is no clear way to determine the ideal value of K, which in turn is sequence dependent and cannot be fixed *a priori*. Torresani *et al*. also argue that these problems will be more relevant as the $rank$ of the data increases, as by allowing more deformation modes there will be ambiguity and more ways in which the model can overfit. Thus, this model has been successfully applied mostly in non-rigid motions that require a relatively small value for K (typically 3 to 5), such as a sparse reconstruction of a human face, but cannot provide satisfactory reconstructions of motions with higher degrees of deformation [34].

After a long period where the low-rank shape basis model dominated the literature, recently other models have been proposed for NRSfM in order to tackle its limitations. In this section we will describe these different approaches and analyse their properties.

Low-rank trajectory basis

In the low-rank shape basis model of Bregler *et al*. [18] the low-rank constraint is applied to the set of 3D coordinates of the points which compose a non-rigid object to constrain and relate their positions in space. What Akhter *et al*. [6] proposed was that the low-rank constraint could be applied not to the spatial configuration of the object, but instead to the temporal evolution of its 3D points (*i.e*. its trajectories). While in Bregler *et al*.'s formulation [18] at each instant in time the 3D position of every point is described as a linear combination of a shape basis, in Akther *et al*.'s approach [6],

for each point its 3D position at each instant is a linear combination of *trajectory basis*, which spans the F images of the observed sequence. Formally, the data matrix is now modelled as

$$\tilde{\mathtt{W}}_i = \mathtt{R}_i \left(\sum_{d=1}^{K} a_{id} \Theta_d \right). \tag{2.16}$$

This model is just a dual representation of the data that does not provide any further insight into solving the problem. However, the great advantage of this method is that there are already successful basis representations for temporal signals (*i.e.* trajectories). If the basis is known, not only does the trilinear problem reduce to a bilinear one, but the need to recompute the basis for every different sequence disappears. In their work, Akther *et al*. [6] have used the Discrete Cosine Transform (DCT) as the basis representation for each coordinate of the point trajectories. This choice of basis makes an implicit assumption of temporally smooth point trajectories, as the bases are ordered by increasing magnitude in the frequency domain. This approach does not remove the need to specify K, which in this case controls which frequencies of the DCT basis are used. Empirically, in most observed signals, the importance of the DCT components decrease as their frequency increases. However this is not necessarily true when modelling the trajectories of highly deformable points where the choice of a low value for K would result in oversmoothing, while overshooting K will, as is the low-rank basis shape cases, lead to overfitting.

Park *et al*. [70] studied the reconstruction ambiguities of this model and showed that there are less ambiguities when the camera motion does not lie in the same subspace of the object motion. Given these properties, they showed how the 3D reconstruction would improve if, for instance, the input images where taken from different cameras in a variety of locations, and then those images were ordered temporally by using the timestamp provided by the cameras. This would effectively be equivalent to a single camera with a very erratic motion satisfying their assumption regarding the object motion.

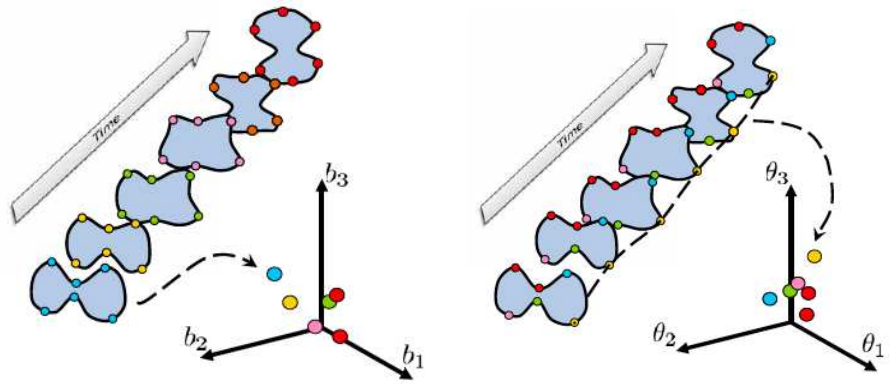


Figure 2.6: Example of the duality of the trajectory and shape basis. Left: the representation of each shape as a point in the shape basis space. Right: the representation of the trajectory of each point over the whole sequence as a point in the trajectory basis space. Figure courtesy of Sohaib Khan and Yaser Sheikh.

Besides removing the need to compute the basis, [6, 70] have the advantage of modelling each point independently, which in practice allows these methods to handle a wider range of motions, such as the motions of human articulated sequence, without violating the low-rank assumption. However, the aforementioned implicit smoothness assumption in the choice of DCT basis limits the reconstructions of this method.

Returning to the low-rank shape basis model, Gotardo and Martinez [44] applied the compact DCT representation to the time evolving shape basis coefficients in $\mathtt{A}_i$ (see Equation 2.16). This combined formulation is very compact, and results in smoothly varying shape basis coefficients which implicitly imposes temporal smoothness on the 3D point trajectories. This formulation also decouples the number of shape basis K to use from the number of DCT basis, allowing for high frequency deformation to be represented without violating the low-rank data assumption. This formulation resulted in a method that outperformed both low-rank basis shape approaches [96, 69] and the trajectory basis approach of [6] (see Figure 2.7).

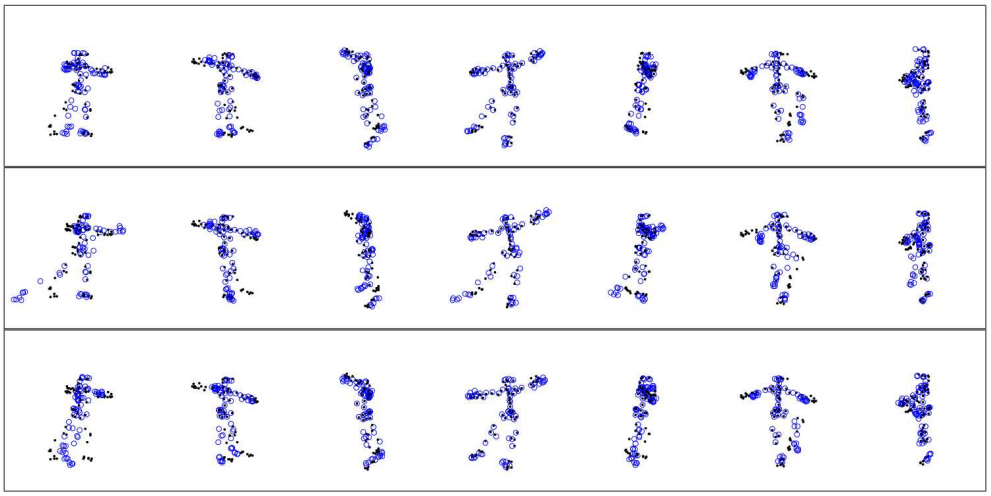


Figure 2.7: Results of several SfM methods in a human motion sequence of a subject dancing. Blue circles represent ground truth data and black dots represent the 3D reconstruction results. Top: Results from Paladini *et al*. [69]. Middle: Results from Akhther *et al*. [6]. Bottom: Results from Gotardo and Martinez [44]. Image from Gotardo and Martinez [44]

Manifold Learning

Rabaud and Belongie [74] also noted that the linear combination of shape basis model severely constrains the possible deformations of the object since they need to lie on a linear subspace of 3D shapes (see Figure 2.8). The authors proposed to relax this constraint and only represent small neighbourhoods of shapes by a linear subspace, with the overall set of possible 3D shapes lying in a smooth low-dimensional manifold of local linear subspaces.

As an initialization, [74] first cluster images that represent similar shapes that can be well described by a single rigid shape. The non-rigid sequences are represented by a temporal succession of rigid shapes denoted as *Rigid Shape Chain*. An optimisation method follows where the motion and shape parameters are estimated while imposing temporal smoothness, and also constraining the 3D shapes to lie on a smooth manifold with dimension K. The manifold is estimated using Locally Smooth Manifold Learning (LSML) [31], although K must be provided.

Zhu *et al*. [109] propose a similar approach where the set of F images are first

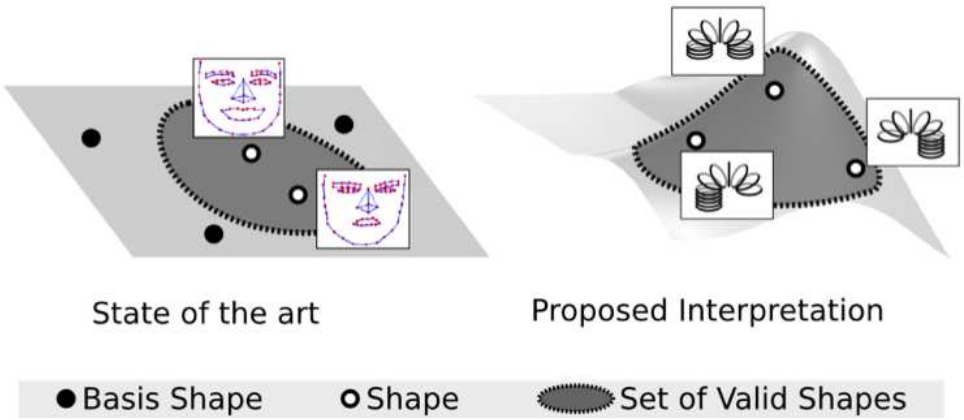


Figure 2.8: Left: Representation of the linear subspace assumed by the low-rank shape basis. Right: Manifold interpretation of Rabaud and Belongie [74]. Figure from Rabaud and Belongie [74].

divided into Q overlapping clusters. Each of these clusters must be ϵ-consistent, *i.e.* the rigid SfM reconstruction for that set of images must result in re-projection error lower than ϵ. In order to lower the computational complexity of forming these clusters, the authors propose a graph traverse method where, starting from initial seeds of ϵ-consistent clusters, images are replaced one by one until all images are part of at least one cluster. Given the multiple possible 3D shape representations that this overlapping set of clusters might propose for a single image, the shapes are clustered using the K-means algorithm into K clusters (with K chosen by the user). Finally [109] uses these K shapes as the shape basis from Bregler *et al.* [18], and perform a non-linear least-squares optimisation to refine the model parameters by minimizing re-projection error (see Section 2.2.1).

Gotardo and Martinez [45] argued that the low-rank shape basis model failed to handle non-linear deformations as this would require a high number for K, which in turn violated the low-rank assumption. To solve this problem, they proposed to apply a *kernel trick*, a common method for non-linear dimensionality reduction, to the NRSfM factorization problem. The 3D configurations of the non-rigid body are then represented as points on a non-liner manifold of dimensions h. The authors claim

that empirically h is usually very small, with a maximum value of 2 in their experiments. Similar to their other work [44], they further compact their representation by constraining the 3D configurations to lie on a smooth trajectory on the non-linear manifold, which is modelled by a linear combination of DCT basis.

Rank reduction via trace-norm minimization

Dai *et al*. [24] proposed to carry out a theoretical analysis of the factorization approaches for NRSfM based on the low-rank basis shape model as they felt the addition of different priors (shape and trajectory bases, temporal and spatial smoothness, inextensibility, etc.) added to solve the NRSfM problem proposed in the literature had prevented a clear understanding of the problem. The main goal of their work was to provide a *prior free* factorization approach for NRSfM. Using the theoretical insight of Akther *et al*. [5] who proved that metric constraints are sufficient to disambiguate the camera motion from the deformable motion of the object, they focused on defining the properties of the metric upgrade matrix. Starting from the non-rigid factorization approach with a given value for K, and defining $\mathtt{E} = \mathtt{G}\mathtt{G}^T$ and $\mathtt{E}^k$ as the k-th column triple of $\mathtt{E}$, the authors showed that $\mathtt{E}_k$ could be found as the solution of the metric upgrade constraints (see Equations 2.7 and 2.8), provided it was positive semi-definite and had $rank - 3$. Due to the numerical instability of the $rank$ *function* [24], the $rank - 3$ constraint was replaced by a $rank$ minimization problem (relaxing it to a trace-norm minimization problem) which can be solved by standard semi-definite programming tools (SDP). After computing $\mathtt{E}^k$, the metric upgrade can be performed and the rotation matrices recovered.

The authors' main insight is that they then go on to estimate $\mathtt{S}$ as the stack of $3F \times P$ matrices containing the F 3D configurations of the object corresponding to image i ($\mathtt{S}_i$) instead of its explicit decomposition into bases and deformation coefficients. Relying on the assumption that $rank(\mathtt{S}) \leq 3K$ and given their estimate of $\mathtt{M}$, $\mathtt{S}$ is recovered by

solving $\tilde{\mathtt{W}} = \mathtt{MS}$ subject to a rank minimization (relaxed as trace-norm minimization) of $\mathtt{S}$. The authors turn to a result from the compressed sensing community which proves that this minimization can be achieved via the the Moore-Penrose pseudo-inverse of $\mathtt{M}$, such that $\mathtt{S} = \mathtt{M}^T(\mathtt{M}\mathtt{M}^T)^{-1}\,\tilde{\mathtt{W}}$.

As an additional result, Dai *et al*. [24] showed that if $\mathtt{S}$ is reshaped into an $F \times 3P$ matrix $\mathtt{S}^\sharp$, the rank of this new shape matrix is now at most K. This fact is used in a new rank minimization problem which will further constrain the shape matrix. However the size of this SDP problem now depends on P and F, which for large numbers can make the computational cost prohibitive.

Although the authors claim their method to be prior free, in fact they rely on the same idea of Bregler *et al*. [18] that non-rigid objects can be globally explained by low-rank matrices. In their experiments, Dai *et al*. [24] have mostly compared against methods which rely on similar principles [104, 96, 6, 69] and have not explored how their method copes with stronger local deformations. Even though [24] outperforms [104, 96, 6, 69], their approach is limited to the cases were the low-rank assumption is known to provide good 3D reconstructions.

2.2.3 Piecewise Approaches

One of the limitations of the low-rank shape basis model is the breakdown in performance when the degree of deformations is increased. To cope with stronger deformations the $rank$ of the basis must be increased, which can quickly lead to overfitting. To tackle these limitations, a recent trend in computer vision has been the proposal of methods that rely on modelling the motion of local regions of points lying on a rigid object, reconstructing them, and later merging these reconstructions into the global non-rigid motion. The intuition of these methods is that local motion is more constrained than global motion, and can be in fact quite similar between objects even if the complexity of the global motion is entirely different. We will now describe the

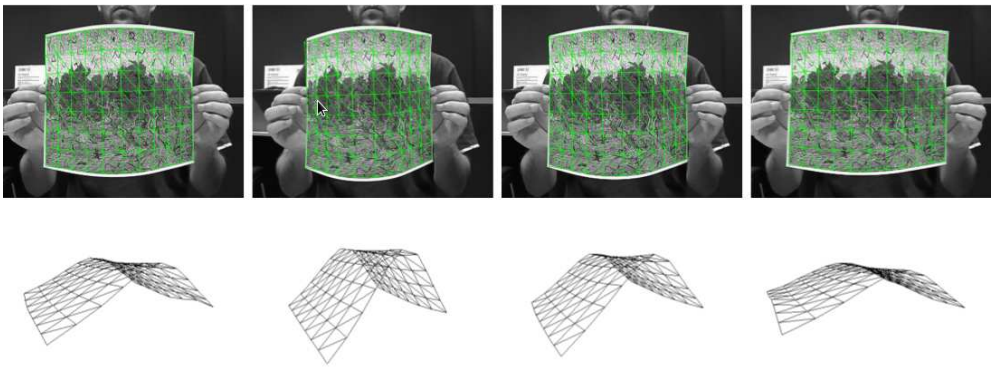


Figure 2.9: Results of Varol *et al*. [100], showing the mesh that is fit to the 3D point cloud and its re-projection on the original images. Image from Varol *et al*. [100].

different piecewise approaches that have been proposed by the computer vision community.

Two-frame piecewise planar reconstruction: The first piecewise approach to NRSfM was proposed by Varol *et al*. [100] where a piecewise planar model was adopted to describe non-rigid motion. The regions were chosen by regular division of the surface into overlapping patches. The approach is valid for a pair of images observed by a calibrated perspective camera. [100] fit a homography to point correspondences in corresponding patches in two images. Since the patches are reconstructed independently there is a need to merge these reconstructions into a single global surface. Points belonging to multiple planes are used to resolve the reconstruction ambiguities and to 'stitch' the patches together. As [100] works in the calibrated perspective camera scenario, imposing 3D consistency is equivalent to solving the depth-scale ambiguity between the independently reconstructed patches.

Varol *et al*. [100] note that in the presence of a sequence of images it is useful to have a consistent representation of the object along the sequence. This is done by fitting a mesh to the 3D point cloud at every frame. The mesh regularises the reconstruction by performing spatial smoothness. Temporal smoothness is added by formulating their optimisation approach over all the frames (see Figure 2.9).

[100] showed how local planar modelling can lead to plausible reconstructions of more complex global motions. It has the advantage of only requiring point correspondences between two frames, although it provides smoother results if temporal consistency is added. On the downside, this is done in a post-processing step where a mesh is added to the set of two-frame reconstructed point clouds, which still results in some flickering. Additionally, the planar regions are chosen manually, raising the question of which division is best. While the simplicity of the planar model is an advantage, results shown in Figure 2.9 hint that it might be too simple, as the reconstructed surface is visibly piecewise planar.

'Soup' of rigid triangles: Taylor *et al*. [90] formulated their reconstruction method around the same idea that even complex non-rigid motion could be locally modelled as rigid planes. In their attempt to model very local motion, Taylor *et al*. [90] restricted their local models to sets of rigidly moving triangles. In terms of choice of local model, this method can be seen as the limiting case of the piecewise planar approach of Varol *et al*. [100], where planes are minimally defined by three non-collinear points. However, [90] does not focus on the two-frame reconstruction problem, and works instead on the framework of P points tracked along F images viewed by an orthographic camera. Their reconstruction results in a set of independently reconstructed triangles with overlapping edges with neighbouring triangles, which the authors named a *triangle soup*. Later these triangles must be merged into the global 3D reconstruction of one or multiple deforming objects.

One of the contributions of [90] was to formulate the reconstruction problem as the recovery of the length of the edges of the triangles instead of directly solving for the 3D positions of its vertices, in what the authors named the Projected Length Equation. The advantage of this formulation is that this set of equations can be solved linearly. However, to overcome noise in the measurements, the edge lengths and vertex positions are used as the initialization of a non-linear least-squares formulation that minimizes

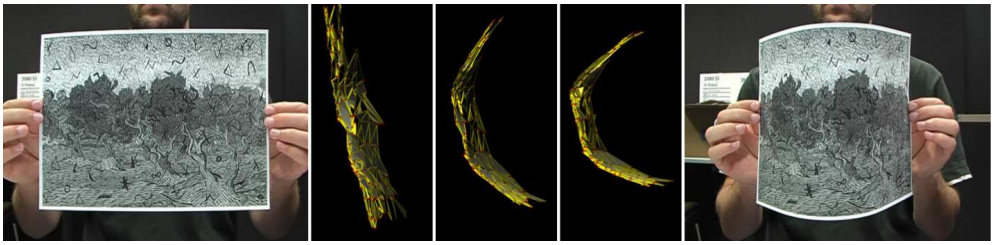


Figure 2.10: Results from Taylor *et al*. [90] showing a top view of their reconstruction. Image from Taylor *et al*. [90]

re-projection error over all the images while refining the reconstructions.

To conform with the locality constraint, the set of triangles to reconstruct is established by 2D Delaunay Triangulation over all F images in the sequence. To further refine the reconstruction, a rigidity test on the triangles is performed, with triangles for which the re-projection error exceeds a certain threshold being discarded. However a re-projection error threshold does not eliminate triangles that overfit, and so two other criteria are used: no angle on the triangle should be less then 10 degrees; and no edge length should be greater then 2.5 times the median length of all the reconstructed triangles.

As triangles are chosen by Delaunay Triangulation they will naturally share vertices, which provides a sufficient constraint to merge them into the overall object reconstruction. However, under orthography, each triangle can only be reconstructed up to a reflection along the camera axis direction, and a translation along that same axis. While solving for the translational component is trivial, solving the reflection ambiguity such that all triangles are consistent is a NP-hard problem. A heuristic greedy solution is then proposed that flips the triangles according to two criteria: the angle between overlapping triangles should be the smallest of the two possible angles; the 3D pose of each triangle should change as little as possible between consecutive images.

The advantage of this method is that the division into local regions derives naturally from the choice of local model. However, its reliance on the limit case of triangular patches leads to an oversegmentation of the objects, increasing the computational load

of reconstruction and, more importantly, of the registration process of the triangular patches. Moreover, it relies on a good balance between the distribution and density of the point tracks, in order to provide a local enough triangle to be well modelled as rigid, but not so local that the triangles become too small and numerous, overfitting to the data.

Piecewise planar weak template reconstruction: In similar spirit to Varol *et al*. [100] and Taylor *et al*. [90], Collins and Bartoli [22] proposed an alternative locally planar approach. Differently from Varol *et al*. [100] their algorithm takes as input an image sequence where P points are tracked along F frames (although missing tracks are allowed), and estimates which regions are best described by planar models instead of relying on manually defined patches. Contrary to [100] and [90], this method does not rely on overlapping areas for the reconstruction.

The segmentation of point tracks into different regions is solved with an MRF-based segmentation approach which clusters points that move according to the same affine motion. This step also allows outlier rejection, where tracks that do not conform with any of the regions are discarded. This results in regions that are not of pre-defined or triangular shape, but have a free shape that is determined by the input data.

Similarly to other piecewise approaches [100, 90], each patch is reconstructed independently by minimizing the re-projection error of the rigidly moving plane. Collins and Bartoli propose a novel closed-form solution to planar motion under orthography, which describes the recovery of the 3D configuration of a plane as a metric upgrade problem, similar to what was described in Section 2.1. Given a relaxation of the non-linear nature of the metric upgrade (although their problem is different, the equations are similar to Equations 2.7 and 2.8), the problem simplifies to a set of polynomial equations that can be solved efficiently. The reflection ambiguity of reconstructing a plane under orthography is treated in a similar way to [90], where the configuration that yields lower deformation and better temporal consistency is preferred. Addition-

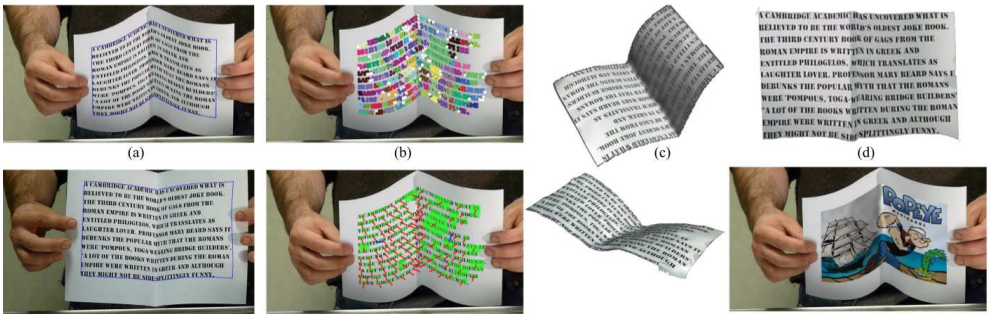


Figure 2.11: Representation of the results from [22]. (a) The region of interest as a blue bounding box. (b) top, each colour represents the points belonging to a given planar region; bottom, the normals of the planar regions marked in red. (c) Densified reconstruction by fitting a smooth surface to the planes; (d) retexturing over the initial image. Figure from Collins and Bartoli [22]

ally, a frame where the configuration is known is used to disambiguate the overall flip ambiguity. Finally, a mesh is fit to the data (similarly to [100]) and its bending energy minimized in order to smooth the results from the individual patches into a consistent surface, and to allow the reconstruction of points that were not directly tracked.

[22] improves over Varol *et al*. [100] by providing a closed-form solution to the multi-frame planar reconstruction problem. Their method also uses an MRF formulation for the choice of planar regions, including an outlier rejection step. The main difference from [22] and the previous piecewise approaches is that these regions do not share any points. Surface consistency is imposed by fitting a mesh to the planar pieces and minimizing the deformation energy. This is however not enough to solve the reflection ambiguity along the Z axis, and [22] requires a manual input a disambiguated frame.

2.3 Template-based deformable surface reconstruction

For completeness, we will now describe the state of the art of the problem of template-based deformable surface reconstruction. This problem is closely related to NRSfM. Like NRSfM, this problem aims at recovering the 3D description of a a deformable

object given an image (or set of images) of a deformed configuration of that object. The difference is that in this problem it is assumed there is a *reference image* I_{ref} for which the corresponding 3D shape of the object is fully known. Given the difficulty to acquire a reliable 3D shape template of a generic object, this family of non-rigid reconstruction methods is generally applied to planar objects such as paper or cloth, as their full geometry can be easily recovered in a fronto parallel image. Template-based approaches allow to reconstruct non-rigid objects using only a pair of images, with many successful approaches being proposed to this day [80, 83, 81, 82, 72, 21].

Given an input image I, the corresponding 3D configuration is estimated by first computing P correspondences between I and I_{ref}. Reconstruction is then achieved by finding the transformation to the 3D template configuration that minimizes re-projection error on image I. However, as in the NRSfM problem, this problem is ill-posed as there are several surfaces which can generate the same 2D projections. Therefore, prior constraints are also needed to disambiguate this problem. Although mostly formulated as a two image problem, these methods can be extended to work with image sequences. A common way to represent the surfaces in the template-based reconstruction is to use a mesh description [83, 82, 80, 72, 21, 8].

One of the priors most commonly used to constrain template-based reconstruction is the assumption of surface inextensibility [80, 72, 21]. In such cases, it is also very common to assume smooth surface deformations to better constrain the problem [21, 72]. Perriollat *et al*. [72], proposed a non-linear least-squares method that uses the thin-plate splines (TPS) as the surface representation. The smoothness deformation constraint was applied by minimizing the bending energy of the shape, which can be easily computed as the second order derivative of the TPS. In a similar approach, Brunet *et al*. [21] represented the surface as a free-form deformations (FFD). Their optimisation method imposed isometry on the solution by imposing an orthonormality constraint on the columns of the Jacobian of the deformation matrix, which was evaluated on a discrete set of points on the parametric surface. Smoothness was

applied by adapting the bending energy to the FFD parametrization of the surface.

In a slightly different approach, Salzmann *et al*. [80] argued that smoothness assumptions on the surfaces prevented the recovery of more complex deformations and thus proposed a method based on temporal consistency over a longer image sequence. In their work, shapes were represented as meshes and the temporal constraint acted on the mesh edges, requiring their orientation to remain similar between consecutive images. [80] formulated this problem as a convex Second Order Cone Programming (SOCP). This change of constraint allowed the recovery of creased surfaces, in contrast with the smooth reconstructions usually achieved with related methods. However this approach still relied on long tacks over the video sequence.

Using the insight of local surface description similar to the one used by the Piecewise NRSfM approaches (see Section 2.2.3), Salzmann *et al*. [83] suggested that template-based reconstruction could improve its accuracy if a model of surface deformation was known. Keeping the representation of the object as a mesh, Salzmann *et al*. [83] learned the possible local deformations of a given surface, using a MoCap system to recover the ground truth 3D positions of a set of tracked points. These points were placed in a grid like pattern over the surface to allow an easy description of the deformations in terms of a mesh.

The temporal constraints of [80] were later replaced by geometric constraints by Salzmann *et al*. [81] which allowed to formulate the template-based reconstruction problem as a two-frame problem. Assuming the reconstructed surfaces to be inextensible, [81] formulates the inextensibility prior by restricting the euclidean distance of neighbouring points to be their geodesic distance, which is measured directly from the fronto-parallel reference image. This formulation results in a set of quadratic equations which can be solved in closed-form. This constraint limits the applicability of this method, as surfaces can only satisfy it if they are deforming smoothly.

Salzmann and Fua [79] later extended [81] to handle surfaces with sharp creases. This is achieved by proposing the inextensibility constraint as an inequality constraint,

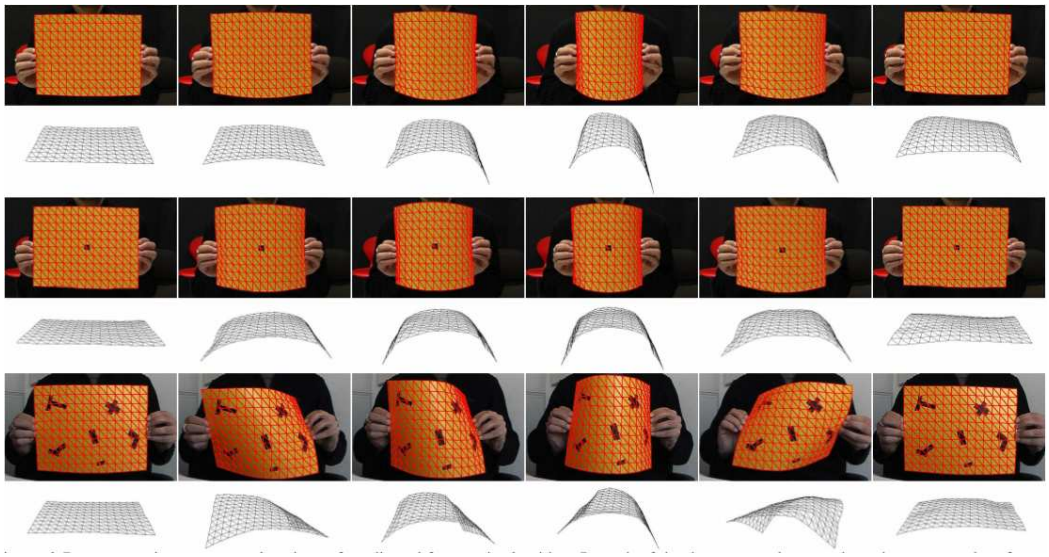


Figure 2.12: Results from [83]. Top: A solution achieved without using the shape constraints of the learned models, which are plausible but not necessarily correct. Middle and Bottom: Reconstruction with the constraint of the local learned models. Reconstructions now match what is observed, even for low textured surfaces. Figure from Salzmann *et al*. [83].

where the euclidean distance of neighbouring points should be less or equal to their geodesic distance. This formulation also resulted convex, although it relied on a heuristic where the depth of the points should be maximized to prevent the surface from collapsing to zero.

Recently Bartoli *et al*. [8] proposed to study the well-posedness of the template-based reconstruction problem. The authors considered the popular isometric surfaces case, and the conformal surface case, which is the family of surfaces that preserve their angles during deformation. Conformal surfaces have been shown to be a good approximation to elasticly deformable surfaces [63]. Bartoli *et al*. [8] showed that there are in fact analytical solutions to the template-based reconstruction problem for both the isometric and conformal case. While a single solution exists for the isometric case (for both the developable and non-developable case), there is a discrete set (at least two) solutions in the conformal case (see Figure 1.7 for results).

Template-based 3D reconstruction is now a well understood problem for which

robust convex formulations exist. However, the need to know the full 3D geometry of the objects is a very strong assumption which has limited the application of these methods mostly to planar surfaces. In general cases, computing a 3D template is difficult or even impossible. This limitation contributes to the attractiveness of NRSfM approaches, which are template-free.

2.4 Articulated motion reconstruction

Articulated motion has been often addressed in the literature as a special case of non-rigid motion. In such case, objects are seen as a set of links joined by articulations. While each link is considered to move rigidly, their motion is dependent due to the articulations joining them into a single kinematic chain [106, 98, 69]. However, other NRSfM approaches presented in Section 2.2 do not distinguish between articulated and deformable motion and reconstruct them in the same framework. [6, 70, 44, 24].

Given the rigid link assumption, A-SfM methods stem from the Tomasi and Kanade factorization approach [93] and extend it to the articulated case. However such approaches require an initial motion segmentation step to divide the object into its constituent parts [106, 98, 69]. Additionally, a classical application of A-SfM methods is human motion reconstruction, approximating it as a set of rigid articulated links. Thus, in this section we will discuss not only existing A-SfM methods but also related work in human motion modelling and reconstruction.

2.4.1 3D pose estimation

The problem of 3D pose estimation from a monocular video sequence is an important one and evidence of this is the large number of works that have addressed it in recent years. An exhaustive review is out of the scope of this thesis, but we refer the reader to [39] for a more complete overview. Two broad classes of strategies have

been used for 3D pose inference: *Generative (top-down)* algorithms optimise a cost function to align an appearance based 3D model with image features [86, 19]; *Discriminative (bottom-up or recognition-based)* methods [2] use training sets of (pose; image) pairs to recognise the pose in a specific image. While generative approaches require prior knowledge of a 3D kinematic model and often require manual initialisation, discriminative methods are dependent on the amount and quality of the training data.

2.4.2 Motion segmentation

Motion segmentation is a particularly challenging problem in the case of articulated motion due to the dependencies between the linked parts. The original solution to the multi-body segmentation problem of Costeira and Kanade [23], based on rigid factorization of Tomasi and Kanade [93], was influential but unable to solve problems containing dependent motions. This was remedied by Zelnik and Irani [107], who built an affinity matrix from the data and used its dominant eigenvectors to separate dependent motions. However, it performed poorly in the presence of articulated motion. The GPCA algebraic framework by Vidal *et al*. [101] can also deal with dependent subspaces and missing data. However, in practice it cannot be applied to more than a few subspaces as the number of required samples grows exponentially with the number of subspaces.

Concerning A-SfM methods, Tresadern and Reid [98] used a RANSAC [94] approach to segmentation, Yan and Pollefeys proposed a segmentation algorithm specifically designed to tackle the articulated motion case [106]. A set of linear subspaces is estimated through local sampling and an affinity matrix is built computing the principal angles between them, followed by spectral clustering to give the segmentation result. Despite outperforming all other motion segmentation algorithms in the cases of articulated motion, this algorithm is highly dependent on the correct detection of the

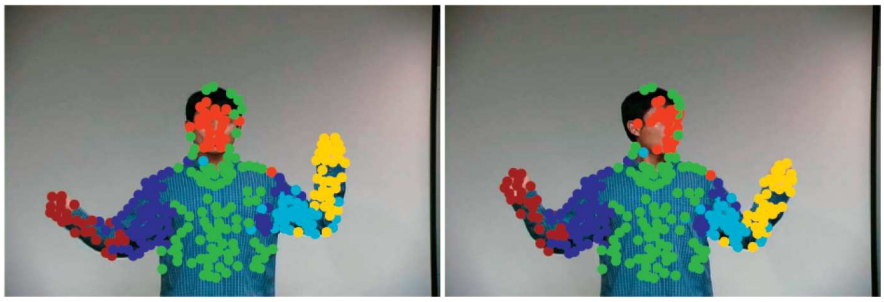


Figure 2.13: Example of the motion segmentation from [106] on a sequence of a man waving his arms. Figure from Yan and Pollefeys [106].

rank of the trajectories, and consequently is sensitive to noise [106].

2.4.3 Articulated structure from motion

Articulated structure from motion (A-SfM) algorithms model such motion as a set of intersecting motion subspaces — the intersection of two motion subspaces implies the existence of a connection between the two corresponding parts. After segmentation, articulation constraints are imposed during factorization to recover the location of joints and axes [98, 106]. While Tresadern and Reid [98] only deal with articulated pairs, Yan and Pollefeys go further [106], estimating the kinematic chain of articulated objects with a more complex structure by building the minimum spanning tree from the segmented subspaces. Factorization is first used to recover the shape and motion of the individual parts, then joints and axes are calculated and used to combine parts into a single coordinate system, and recovering the articulated shape and motion as a whole. Ross *et al*. [76] instead propose a probabilistic approach to learn the structure of an articulated object while inferring its pose given a time series of 2D feature tracks. This method generally places joints in the middle of segments, rather than at endpoints, and has difficulty to recover from a poor initialisation.

These A-SfM methods offer an attractive solution to 3D pose estimation since they are model-free and do not require any training data. However they suffer from a number of weaknesses: the quality of the motion segmentation step is critical for a suc-

cessful 3D reconstruction– misclassified points can lead to large errors in the motion and shape estimates; motion segmentation algorithms on the other hand either require the number of constituent object parts to be known in advance or are sensitive to its mis-estimation via the detection of the rank of the trajectories; missing data cannot be dealt with; and finally, their demonstrations are on simple articulated motions, crucially never on a full body. These algorithms focus less on the full 3D reconstruction of the body and more on estimating the location of joints and articulation axes to estimate the skeleton structure.

2.5 Proposed approach

In this thesis we follow the emerging trend of piecewise NRSfM methods and propose a principled solution to this problem. Instead of relying on a manual division of patches like [100], or in a model derived division like [90], we provide a formulation for simultaneous patch division and reconstruction, where both steps solve the same geometric cost – the image re-projection error. We propose a division into overlapping patches, which is ensured by our formulation, and provide an approach to merge the independent reconstructions by enforcing the 3D consistency of overlapping points. In addition, we provide our own local model – the Quadratic Deformation (QD) model – and show how this formulation scales towards dense NRSfM. Finally, we show how our principled piecewise approach can be used in the problem of articulated SfM, where each segment of the articulated structure is modelled as a rigid body.

Chapter 3

Quadratic Deformation Model for NRSfM

As discussed in Chapter 2, most approaches to NRSfM [18, 96, 17, 104] modelled deformations using the low-rank shape basis model. This model has provided plausible reconstructions in the case of small deformations of objects with a dominant rigid component, such as the human face [96, 7]. However, not all types of deformations can be described by this simple shape model. In particular if the shapes are undergoing stretch, bending or twist deformations, a different model is needed to represent the non-linearity of the motion. A low rank linear shape model would account for the non-linear deformations simply by adding new basis shapes and this results in overfitting and incorrect depth estimates. Only recently have other models been proposed to tackle non-linear deformations [74, 45]. In this chapter we introduce the Quadratic Deformation (QD) model which is both compact and physically grounded and can encode the non-linear variations that appear in more complex motions.

The QD model was first proposed as a description of non-rigid deformation by Müller *et al*. [66]. In this computer graphics work, the QD model was proposed as a point-cloud based method with geometric grounds, which gives a natural and versatile description of non-rigid 3D objects as a second order polynomial. The estimation of

deformation coefficients was stable and computationally efficient. This work showed how the model can be applied in a piecewise fashion, increasing the range of non-rigid motions it can handle.

Taking advantage of the natural deformations the QD model describes, Park and Hodgins [71] proposed to use the QD model in a piecewise fashion to model the deformable motion of human skin. In their work, they capture the 3D motion of a human subject using a set of markers sparsely distributed on the subject's body, and used the QD model to reconstruct the skin deformations in higher detail. Their results show that with the QD model it is possible to recover muscle deformations and high acceleration skin motions.

Inspired by the work of Park and Hodgins, our previous work [35] used the QD model to describe human soft-tissue deformation. In biomechanics, the human skeleton is often modelled as set of articulated rigid segments (see Chapter 7). Accuracy in estimating the motion of the articulated skeleton may be affected by the non-rigid motion of the soft-tissue. [35] attempted to remove these soft-tissue artefacts by recovering them with the QD model.

Given the success of the QD model in describing non-rigid deformations in 3D, in this chapter we formulate how it can be used within a NRSfM framework to perform non-rigid 3D reconstruction from an image sequence. We analyse the QD model and its deformation modes, and explain why it is suitable for non-rigid motion description.

3.1 Quadratic Deformation Model for Non-Rigid Bodies

As described in Section 2.1, a rigid body composed of P points is represented as a $3 \times P$ matrix $\mathtt{S}_r$ containing the 3D coordinates of those points in the object reference

frame:

$$\mathtt{S}_r = \begin{bmatrix} X_1 & X_2 & \dots & X_P \\ Y_1 & Y_2 & \dots & Y_P \\ Z_1 & Z_2 & \dots & Z_P \end{bmatrix}. \tag{2.2}$$

The image of a rigid body under orthographic viewing conditions is given by the orthographic projective equation

$$\mathtt{W}_i = \mathtt{\Pi R}_i\mathtt{S} + \mathtt{T}_i, \tag{2.3}$$

where $\mathtt{\Pi}$ is the orthographic projection matrix

$$\mathtt{\Pi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{2.1}$$

$\mathtt{R}_i$ is a 3×3 rotation matrix that aligns the reference frame with the correct pose, and $\mathbf{t}_i$ a 2-vector accounting for image translation. The orthographic image of a non-rigid body can be derived in a similar way, except the shape of the object is allowed to vary from frame to frame:

$$\mathtt{W}_i = \mathtt{\Pi R}_i\mathtt{S}_i + \mathbf{t}_i. \tag{2.9}$$

Here, the QD model is a parametrization of $\mathtt{S}_i$ which encodes the deformations that the shape can undergo. More specifically, the model is created by augmenting the shape matrix with extra terms containing the quadratic and cross-term products of its entries. The shape of a body can be deformed by manipulating some coefficients which act on the augmented coordinates. We now analyse the nature of this coordinate augmentation and the properties of the QD model. We define the augmented shape

matrix for the QD model as:

$$\mathtt{S}_q = \left[\begin{array}{cccc} X_1 & X_2 & \dots & X_p \\ Y_1 & Y_2 & \dots & Y_p \\ Z_1 & Z_2 & \dots & Z_p \\ \hline X_1^2 & X_2^2 & \dots & X_p^2 \\ Y_1^2 & Y_2^2 & \dots & Y_p^2 \\ Z_1^2 & Z_2^2 & \dots & Z_p^2 \\ \hline X_1Y_1 & X_2Y_2 & \dots & X_pY_p \\ Y_1Z_1 & Y_2Z_2 & \dots & Y_pZ_p \\ Z_1X_1 & Z_2X_2 & \dots & Z_pX_p \end{array}\right] = \left[\begin{array}{c} \mathtt{S}^{(\mathtt{L})} \\ \hline \mathtt{S}^{(\mathtt{Q})} \\ \hline \mathtt{S}^{(\mathtt{C})} \end{array}\right], \tag{3.1}$$

where $\mathtt{S}^{(\mathtt{L})}$ is the $3 \times P$ linear shape matrix which contains the 3D coordinates of a given point cloud in the object referential, while $\mathtt{S}^{(\mathtt{Q})}$ and $\mathtt{S}^{(\mathtt{C})}$ are $3 \times P$ matrices which contain respectively the quadratic and cross-term products of those 3D coordinates. Since we augmented the shape matrix to a $9 \times P$ matrix, to keep dimensions consistent we must define a 3×9 transformation matrix which maps the system back to 3D. We define this transformation as the Quadratic Deformation transformation matrix $\mathtt{A}_i$:

$$\mathtt{A}_i = \left[\begin{array}{ccc} \mathtt{L}_i & \mathtt{Q}_i & \mathtt{C}_i \end{array}\right], \tag{3.2}$$

where $\mathtt{L}_i$, $\mathtt{Q}_i$ and $\mathtt{C}_i$ are 3×3 matrices. As will be shown in more detail in Section 3.2, these matrices contain the coefficients which control the deformation modes allowed by the QD model. These matrices will be named respectively the linear, quadratic and cross-term deformation coefficient matrices. Finally the shape matrix at frame i can be defined in terms of these new parameters as:

$$\mathtt{S}_i = \mathtt{A}_i\mathtt{S}_q = \left[\begin{array}{ccc} \mathtt{L}_i & \mathtt{Q}_i & \mathtt{C}_i \end{array}\right] \mathtt{S}_q. \tag{3.3}$$

Notice that the shape matrix $\mathtt{S}_q$, which encodes the augmented coordinates, is fixed for all the frames in the sequence while the deformation matrix $\mathtt{A}_i$ varies from frame to frame. As Equation 3.1 shows, $\mathtt{S}^{(\mathtt{Q})}$ and $\mathtt{S}^{(\mathtt{C})}$ are simply functions of $\mathtt{S}^{(\mathtt{L})}$. If $\mathtt{L}_i = \mathtt{I}_{3\times3}$, $\mathtt{Q}_i = \mathtt{0}_{3\times3}$ and $\mathtt{C}_i = \mathtt{0}_{3\times3}$ for all frames i, we revert to the case of rigid body motion with $\mathtt{S}^{(\mathtt{L})}$ taking the role of the rigid shape. This observation is indeed important and implies that $\mathtt{S}^{(\mathtt{L})}$ encodes the shape of the object that will be deformed into the different configurations allowed by the model. For this reason, we will also refer to $\mathtt{S}^{(\mathtt{L})}$ as the *rest shape* matrix (see 3.3.2), although this is just an intuition and its meaning should not be taken literally.

3.2 The Quadratic Model Deformation Modes

To simplify the analysis of the deformation modes, we will drop the frame dependency index i and will refer to the deformation matrix simply as $\mathtt{A}$. For a better understanding of the model we will analyse the three parts ($\mathtt{L}$, $\mathtt{Q}$ and $\mathtt{C}$) independently. For this analysis it is helpful to define a "default" state of the model from which we will vary the deformation coefficients, and study the corresponding deformations. In line with the intuition of the rest shape, we define this "default" state as the one that corresponds to rigid body motion, where $\mathtt{L} = \mathtt{I}_{3\times3}$, $\mathtt{Q} = \mathtt{0}_{3\times3}$ and $\mathtt{C} = \mathtt{0}_{3\times3}$. We will refer to the different coefficients as L_{mn}, Q_{mn} and $C_{mn} \forall m, n = 1, 2, 3$. To better understand the effects of this model we apply these transformations to a generic augmented point $\mathbf{x} = [x,\, y,\, z,\, x^2,\, y^2,\, z^2,\, xy,\, yz,\, zx]^T$, and visualize the corresponding deformations of points lying on a 10×10 grid on the x,y-plane. By deforming a a planar object it is easier to to infer the deformations modes of the QD model, and the generalization into 3D point clouds is straight forward.

3.2.1 Linear Deformation Coefficients

The matrix $\mathtt{L}$ accounts for affine deformations. To prevent degenerate cases we force $\mathtt{L}$ to be full-rank. A full-rank 3×3 matrix can always be decomposed into the product of a rotation matrix and an additional transformation matrix (*e.g.* QR factorization, LQ factorization, Polar decomposition, etc. [43]). As shown in Equation 3.10, the rotation is already modelled explicitly by the camera model. Therefore in the QD model formulation we define $\mathtt{L}$ as having no rotational component *i.e.* the rotation matrix of the decomposition is chosen to be the 3×3 identity matrix to avoid over-parametrisation. All choices of matrix decompositions are equivalent in the sense that they represent the set of all possible 3×3 full-rank matrices. However, the polar decomposition is widely used in physics due to the elegant description it provides of the relationship between deformation and rotation terms. We will also use this description in our framework.

Defining $\mathtt{A}_{1:3}$ as the 3×3 full-rank matrix containing the first 3 columns of $\mathtt{A}$, by polar decomposition we say that $\mathtt{A}_{1:3} = \mathtt{R}\,\mathtt{L}$, where $\mathtt{R}$ is a 3×3 rotation matrix and $\mathtt{L}$ is a 3×3 symmetric matrix. We define $\mathtt{R}$ as the camera rotation matrix and $\mathtt{L}$ as the linear deformation matrix in our formulation. We now analyse the kind of transformations which result from modifying these coefficients.

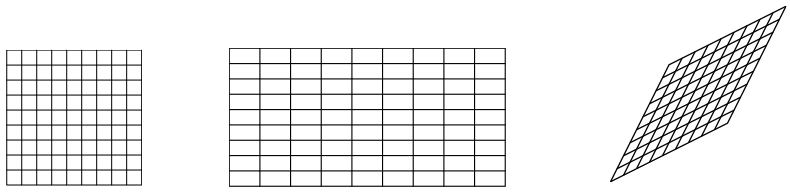
Diagonal coefficients:

Given a generic 3D point $(x, y, z)^T$ and its corresponding augmented coordinates $\mathbf{x} = [x,\, y,\, z,\, x^2,\, y^2,\, z^2,\, xy,\, yz,\, zx]^T$, if we now choose to set the diagonal coefficients L_{nn} to any value, and apply the resulting matrix $\mathtt{L}$ on the augmented point $\mathbf{x}$, the

resulting deformed point will be

$$\mathbf{x}' = \mathtt{A}\mathbf{x} = \begin{bmatrix} L_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_{22} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & L_{33} & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} L_{11}x \\ L_{22}y \\ L_{33}z \end{bmatrix}, \tag{3.4}$$

which means L_{nn} has a *scaling* effect along the n-th axis of the local referential. Figure 3.1 (b) shows an example of the deformations caused by L_{11}.



(a) (b) (c)

Figure 3.1: Illustration of the deformation caused by the coefficients of the 3×3 matrix L on square object. (a) Undeformed square object. (b) Effect of deformation with $L_{11} - 0.5$. (c) Effect of the deformation of $L_{12} = L_{21} = 0.5$. The deformations on the other dimensions for a 3D object can be easily generalised from these examples.

Off-diagonal coefficients:

Changing the value of L_{12} (and consequently of L_{21} since L is chosen to be symmetric) the deformed point $\mathbf{x}'$ will now be:

$$\mathbf{x}' = \mathtt{A}\mathbf{x} = \begin{bmatrix} 1 & L_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ L_{21} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} x + L_{12}y \\ y + L_{21}x \\ z \end{bmatrix}. \tag{3.5}$$

The effect of the coefficient pair $L_{12} \backslash L_{21}$ is a pure *shearing* deformation along the first and second axis of the object as shown in Figure 3.1 (c). The deformation caused by the other two pairs of off-diagonal coefficients are analogous.

3.2.2 Quadratic Deformation Coefficients

When describing a rigid motion the deformation coefficients of `Q` and `C` are set to zero. Therefore, unlike `L`, `Q` need not be full rank. In fact, the deformation coefficients of `Q` can be separated into two groups according to the nature of the deformation mode modelled by them.

Diagonal entries:

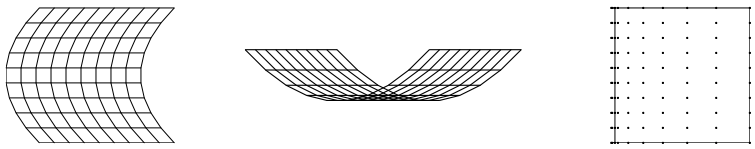
If we change the value of the diagonal coefficients Q_{nn} the effect on the augmented point $\mathbf{x}$ is:

$$\mathbf{x}' = \mathtt{A}\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & Q_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & Q_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & Q_{33} & 0 & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} x + Q_{11}x^2 \\ y + Q_{22}y^2 \\ z + Q_{33}z^2 \end{bmatrix}. \quad (3.6)$$

These diagonal coefficients Q_{nn} act in a similar way to L_{nn} and account for a *non-linear* scaling effect along the n-th axis of the local coordinates. When applying this transformation to the cube shape object shown in Figure 3.2, we note that points where the x coordinate and Q_{11} have the same sign will have a non-linear expansion along the first axis, while points where the x coordinate and Q_{11} have opposite sign will undergo non-linear compression. This non-linear compression can lead to points changing their relative configuration within the object, as points belonging to the outer edges are deformed into the centre of the object at a faster rate than points belonging to interior edges in the underformed case (*e.g.* see Figure 3.2 (c)). Analogous behaviour can be found if we analyse the effects of Q_{22} and Q_{33}.

Off-diagonal elements:

Unlike `L`, there are no symmetry constraints on `Q` and the off-diagonal deformation coefficients can be studied independently. Let us vary Q_{12} and check its effect on the



(a) (b) (c)

Figure 3.2: Illustration of the deformation caused by the coefficients of the 3×3 matrix $\mathtt{Q}$ on square object defined on the x,y-Plane. (a) In the plane bending deformation cause by $Q_{12} = 0.5$. (b) Out of the plane bending deformation caused by $Q_{12} = 0.5$. (c) Example of the interpenetration of the outer points into the inner points caused by Q_{11} highlighted by only displaying the outer edges of the square object and enlarging inner intersection points of the grid. The deformations on the other dimensions for a 3D object can be easily generalised from these examples.

augmented point $\mathbf{x}$:

$$\mathbf{x}' = \mathtt{A}\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 & Q_{12} & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} x + Q_{12}y^2 \\ y \\ z \end{bmatrix}. \tag{3.7}$$

In this case there is a *non-linear scaling* along the first axis of the local coordinate system, only now it depends quadratically on y instead of x. This means that points that lie further from the first axis when measuring the distance along the second axis will have a larger scaling factor, which in turn results in a *bending* motion. There are six independent bending motions, two per axis. An example of such bending motions can be seen in Figure 3.2 (b).

3.2.3 Cross-term Deformation Coefficients

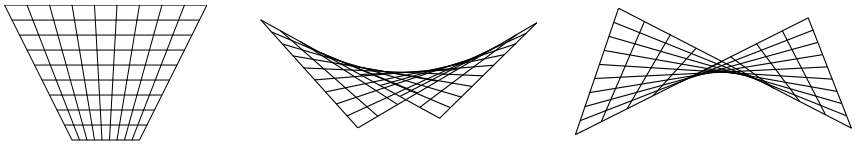
Unlike `L` and `Q`, the cross-terms cannot be divided into diagonal and off-diagonal coefficients. Still, there are also two distinctive families of deformations: those that depend on all three coordinates x y and z (C_{12}, C_{23} and C_{31}), and those that depend on just two coordinates (C_{11},C_{13},C_{21},C_{22},C_{32} and C_{33}).

Dependency on three coordinates:

We consider a single deformation coefficient as an example and infer the role of the other deformations by analogy. In this case, if we vary C_{12} from our "default" configuration the resulting deformation is:

$$\mathbf{x}' = \mathtt{A}\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & C_{31} & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} x \\ y \\ z + C_{31}xy \end{bmatrix}, \tag{3.8}$$

corresponding to a translation of the z coordinate proportional to product of x and y. This type of deformation corresponds to a *twisting* motion of the object and is illustrated in Figure 3.3 (b) and (c).



(a) (b) (c)

Figure 3.3: Illustration of the deformation caused by the coefficients of the 3×3 matrix $\mathtt{C}$ on square object defined on the x,y-Plane. (a) In the plane deformation cause by $C_{11} = 0.5$. (b) Out of the plane twisting deformation caused by $C_{31} = 0.5$. (c) Another view of the twisting deformation caused by $C_{31} = 0.5$. The deformations on the other dimensions for a 3D object can be easily generalised from these examples.

Dependency on two coordinates:

If we now choose to vary C_{11}:

$$\mathbf{x}' = \mathtt{A}\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & C_{11} & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} = \begin{bmatrix} x + C_{11}xy \\ y \\ z \end{bmatrix}, \tag{3.9}$$

it results in a deformation that translates the x coordinate in proportion to the product

of x and y. This deformation mode essentially generates a non-linear scaling along the x coordinate that creates a *compression* motion on one half of the object while *expanding* the other half according to the sign of x and C_{11} (see Figure 3.3).

3.3 Non-Rigid SfM with a Quadratic Deformation Model

Having described how the QD model can be used to encode non-rigid motion in 3D, we show how it can be used as a shape model for 3D reconstruction of non-rigid motion from image measurements. Our approach to NRSfM can be framed in the paradigm of Tomasi and Kanade [93] where a set of P feature points are observed across F images by an orthographic camera, and these measurements are factorised into the product of camera motion and shape matrices. We can simply replace the deformed shape $\mathtt{S}_i$ in Equation 2.9 by the corresponding parametrization of the QD model. Writing it in terms of the i-th frame and j-th point, we have:

$$\mathbf{w}_{ij} = \Pi \, \mathtt{R}_i \, \left[\mathtt{L}_i \, \mathtt{Q}_i \, \mathtt{C}_i\right] \, \mathbf{s}_j \, + \, \mathbf{t}_i, \tag{3.10}$$

where $\mathbf{w}_{ij}$ is the 2D image position of point j in image i, and $\mathbf{s}_j$ is the j-th column of $\mathtt{S}_q$. The translational component can be easily removed by registering the point cloud at every image i to its centroid. We stack all the sub-block matrices for each image i obtaining:

$$\tilde{\mathtt{W}} = \begin{bmatrix} \Pi\mathtt{R}_1 & & & \\ & \Pi\mathtt{R}_2 & & \\ & & \ddots & \\ & & & \Pi\mathtt{R}_F \end{bmatrix} \begin{bmatrix} \mathtt{L}_1 & \mathtt{Q}_1 & \mathtt{C}_1 \\ \mathtt{L}_2 & \mathtt{Q}_2 & \mathtt{C}_2 \\ \vdots & \vdots & \vdots \\ \mathtt{L}_F & \mathtt{Q}_F & \mathtt{C}_F \end{bmatrix} \begin{bmatrix} \mathtt{S}^{(\mathtt{L})} \\ \mathtt{S}^{(\mathtt{Q})} \\ \mathtt{S}^{(\mathtt{C})} \end{bmatrix} = \check{\mathtt{M}} \, \mathtt{A} \, \mathtt{S}_q = \check{\mathtt{M}}\mathtt{S}, \tag{3.11}$$

where $\tilde{\mathtt{W}}$ is the $2F \times P$ measurement matrix registered to the centroid at each frame.

As discussed in Chapter 2, factorization approaches have proved practical in solv-

ing the rigid and non-rigid SfM problems. These methods can be loosely described as a rank-constrained singular value decomposition followed by fifing a metric upgrade matrix which applies the orthonormality constraints to $\mathtt{R}_i$. As discussed in Section 2.2.1, computing the metric upgrade matrix is a difficult problem. While Akther *et al*. [6] proved that a metric upgrade matrix was sufficient to perform NRSfM via factorization, they also stated that the constraints of finding the true solution of that matrix are non-linear, and did not provide a closed-form solution to that problem. In fact, successful NRSfM via factorization always relied on solving a simplified version of the metric upgrade problem with assumptions that did not hold for every case [104, 17].

Drawing a parallel from the low-rank shape basis model used by previous factorization methods and the QD model, we see that although we have the same orthonormality constraints to impose on $\mathtt{R}$, we still need to account for a very specific structure for both $\mathtt{A}$ and $\mathtt{S}_q$. Looking at the structure of the augmented shape matrix $\mathtt{S}_q$ on the QD model, there are relationships between the terms linear, quadratic and cross-terms that could be exploited in constraining such upgrade matrices. However these constraints are also non-linear, for which computing a corrective matrix proves to be hard.

Given the additional difficulty in computing the corrective matrices that arise from the additional constraints in $\mathtt{A}$ and $\mathtt{S}_q$, and the disadvantages of closed-form methods discussed in Chapter 2, we choose to formulate our NRSfM algorithm as a non-linear least squares problem. This formulation allows us to estimate our model parameters explicitly, keeping the desired structure for the model matrices $\mathtt{A}$ and $\mathtt{S}_q$. Additionally, the orthonormality of $\mathtt{R}$ can also be enforced exactly by choosing an appropriate parametrization (e.g. quaternion unit vectors).

3.3.1 Non-linear optimization

We formulate NRSfM with the QD model as a non-linear least-squares optimisation that minimizes the 2D re-projection error of the 3D reconstruction. In particular we

use the Levenberg-Marquardt non-linear least squares algorithm directly exploiting the sparse properties of the Jacobian and Hessian matrices computed at each iteration of the minimization. Starting from Equation 3.10, we can now define the re-projection error as:

$$\mathcal{R}(\mathbf{w}_{ij}, \mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i, \mathbf{s}_j) = ||\mathbf{w}_{ij} - \hat{\mathbf{w}}_{ij}||^2 = ||\mathbf{w}_{ij} - \Pi\mathtt{R}_i(\mathbf{q}_i)\,[\mathtt{L}_i\,\mathtt{Q}_i\,\mathtt{C}_i]\,\mathbf{s}_j - \mathbf{t}_i||^2\,, \tag{3.12}$$

where $\mathtt{R}_i(\mathbf{q}_i)$ indicates that, internally, the rotations are parametrised using quaternion vectors $\mathbf{q}_i$, which are the actual parameters to estimate. As discussed in Chapter 2, one of the most important advantages of using a non-linear minimization scheme to minimize image re-projection error is that the cost function is parametrised explicitly using all the parameters of the QD model. Therefore, any prior information available about the nature of the object being observed that has an effect on the values that the deformation matrices $\mathtt{L}_i$, $\mathtt{Q}_i$ and $\mathtt{C}_i$ can take may be incorporated into the cost function. Similarly to [1], we incorporate temporal smoothness priors over all the parameters:

$$\begin{aligned}\mathcal{R}_\lambda(\mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i) = {} & \lambda_{\mathtt{LQC}}\,||[\mathtt{L}_i\,\mathtt{Q}_i\,\mathtt{C}_i] - [\mathtt{L}_{i-1}\,\mathtt{Q}_{i-1}\,\mathtt{C}_{i-1}]||^2 \\ & + \lambda_{\mathbf{t}}\,||\mathbf{t}_i - \mathbf{t}_{i-1}||^2 \\ & + \lambda_{\mathbf{q}}\,||\mathbf{q}_i - \mathbf{q}_{i-1}||^2 \quad ,\end{aligned} \tag{3.13}$$

where $\lambda_{\mathtt{LQC}}$, $\lambda_{\mathbf{t}}$ and $\lambda_{\mathbf{q}}$ are user defined weights to tune the regularisation. Details on how these parameters were chosen are presented in Section 3.4.

Finally we can combine Equation 3.12 and Equation 3.13 into the final cost to minimize:

$$\underset{\mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i}{\arg\min} \sum_{i,j}^{F,P} \mathcal{R}(\mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i) + \sum_{i=2}^{F} \mathcal{R}_\lambda(\mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i). \tag{3.14}$$

In addition to good regularization terms (*e.g.* the temporal smoothness parameters in

$\mathcal{R}_\lambda$), the likelihood of falling into local minima can be reduced by providing a good initialization to the solution, close to the global minimum. Notice that in our formulation we do not optimize for $\mathtt{S}_q$ as empirically we observed lower 3D errors without modifying its values. In Section 3.3.2 we will explain how we can obtain a good initialization of $\mathtt{S}_q$ that improves our solution.

By keeping the shape matrix $\mathtt{S}_q$ out of the optimized parameters, our problem reduces to a bilinear problem of estimating $\mathtt{R}$ and $\mathtt{A}$. At this point, our problem is similar to [5], where the trilinear problem was reduced to a bilinear problem by assuming known DCT trajectory basis for the points trajectories over time (see Section 2.2.2 for details). However, [5] relies on an approximation of the metric upgrade matrix to correctly factor $\mathtt{R}$. Instead, we will follow our arguments presented in Section 3.3, and opt for a non-linear optimization framework, where the constraints on $\mathtt{R}$ and $\mathtt{A}$ can be imposed explicitly, even though the difficulty in correctly recovering the structure in $\mathtt{S}_q$ has been overcome with our choice for a fixing those terms.

3.3.2 Initialization

As discussed in Chapter 2, the Levenberg-Marquardt algorithm cannot guarantee convergence to the global optimum, relying on the initial estimates being close to the global minimum to avoid falling into local solutions. In our optimisation problem we must provide adequate initial values for the rotation matrices $\mathtt{R}_i$, the deformation coefficients in $\mathtt{L}_i$, $\mathtt{Q}_i$ and $\mathtt{C}_i$, and the shape matrix $\mathtt{S}_q$. As we saw in Section 3.3, the QD model can easily revert back to the rigid model if the deformation matrices are fixed to predetermined values. This also shows that $\mathtt{S}^{(\mathtt{L})}$ can be interpreted as the object's shape without any deformation. In fact, the QD model is idealized as a physically grounded model where a known object is deformed according to the quadratic deformation modes. Prior work in computer graphics where the QD model was used [66, 71] supports this view as both works considered a known 3D configuration as reference from which the ob-

ject is deformed. This implies that a correct 3D reconstruction greatly depends on a reasonable choice for $\mathtt{S}^{(\mathtt{L})}$. In reality, initializing such shape form the data is a hard problem when no more information about the object properties is available. Based on the rest-shape intuition, we propose, for now, to initialize our method assuming that the object will undergo close to rigid motion for the first few frames to recover a good initialization for $\mathtt{R}_i$ and $\mathtt{S}_q$ using a rigid factorization approach such as [64]. We leave the discussion of other scenarios where a good initialization can be achieved to Chapter 4. Finally, the QD model parameters are initialized as $\mathtt{L}_i = \mathtt{I}_{3\times3}$, $\mathtt{Q}_i = \mathtt{0}_{3\times3}$ and $\mathtt{C}_i = \mathtt{0}_{3\times3}$ for every image i.

In the case where a 3D template of the object to reconstruct is available, $\mathtt{S}^{(\mathtt{L})}$ can also be initialized to that 3D shape, making our approach very close to the template-based reconstruction methods presented in Section 2.3. In truth, it could be argued that the framework presented in this chapter can be divided into into 3D rest-shape estimation followed by a a template-based reconstruction. However, it is not the goal of this thesis to develop template-based reconstruction methods, and so we try our best to rely solely on what can be extracted from the data. This allows us to reconstruct a greater variety of sequences, as a 3D template is not often available to be used. Additionally, we do not consider the two-frame reconstruction case, always relying on having stronger temporal information from an image sequence to perform our reconstruction.

3.4 Experiments

We devised a series of experiments to test the robustness and applicability of the QD model to the NRSfM problem. We measured the 3D reconstruction error on synthetic sequences under different circumstances, and compare our approach with other NRSfM methods on a motion capture sequence for which ground truth is available. Finally we show 3D reconstruction results on real image sequences.

3.4.1 Synthetic cylinder sequence

For the synthetic experiments we created a cylindrical object with 70 points which we used as the rest shape for our model. We then applied deformations of increasing maximum strength to the object using the deformation matrices $\mathtt{L}$, $\mathtt{Q}$ and $\mathtt{C}$, with magnitudes ranging from from 0 to 1, starting from the value of the rigid motion case. To account for our model hypothesis we made sure there was enough rigid motion for the first few frames of each sequence, and that both the deformation coefficients and rotation matrices chosen generated a temporally smooth motion. We generated 50 random tests for each level of maximum magnitude level, keeping all the other parameters unchanged. The 3D points were projected onto the image using an orthographic camera model. The three different λ parameters from Equation 3.13 were tuned based on the 3D reconstruction error on the set of synthetic sequences, and an empirical observation of how flexible the object was, to avoid a situation overfitting. After running a batch of tests with the three λ parameters ranging from 10^{-1} to 10^{-5}, the best compromise between accuracy and flexibility was found to be $\lambda_{\mathtt{LQM}} = \lambda_{\mathbf{t}} = \lambda_{\mathbf{q}} = 10^{-2}$.

We compare the results of our new algorithm (Quad) with Torresani *et al.*'s algorithm [96] (EM-LDS) and with a Bundle Adjustment algorithm (BA-Lin) [26], both of which are based on the linear low-rank shape model. For a fair comparison, we initialized BA-Lin with the same parameters for camera and shape matrices that we obtain from our initialization by running [64] on the first 10 images of the sequence in which the object was not deforming. For the EM-LDS method we used its own initialization, as it provided better results.

To compute the 3D error we use the same measure as [105, 96, 69]. Defining $\mathtt{X}$ as the $3F \times P$ matrix containing the 3D ground truth positions of the P points we want to reconstruct, while $\hat{\mathtt{X}}$ is the 3D reconstruction generated by a given method, we compute the normalized reconstruction error as $\left|\left|\mathtt{X} - \hat{\mathtt{X}}\right|\right|_F / \left|\left|\mathtt{X}\right|\right|_F$, where $||.||_F$ is the Frobenius norm.

In Figure 3.4 we show the average 3D reconstruction error as well a box-plot analysis for each of the algorithms. The average error plot was generated after removing the results from tests considered outliers by our statistical analysis (marked as red crosses in the box-plots of Figure 3.4). Our new algorithm outperforms the other methods in two important aspects. First, the box-plots show a lower rate of outliers compared to the other algorithms. With Quad only 3.09% of all the tests are outliers, while with EM-LDS as many as 8.91% were considered outliers and 9.45% with BA-Lin. Secondly, amongst the tests considered as inliers, the average error plot (Figure 3.4 top left) shows that the lowest 3D error was given by our method.

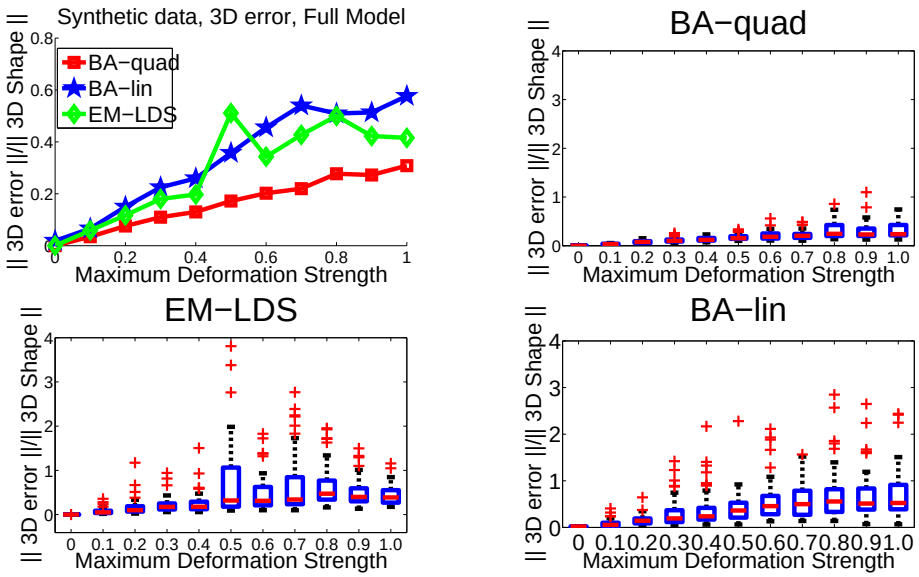


Figure 3.4: Results on synthetic data for varying deformation strengths. Top-left: Average 3D error plot for experiments that converged to a valid solution. Box-plots are provided to illustrate the rate of convergence of the three different algorithms. Notice the high rate of convergence failure of the EM-LDS and BA-Lin algorithms.

3.4.2 Experiments with real deformations from MoCap data

In these experiments we used 3D motion capture data of a water **Woggle** (or swimming noodle) which is a long and thin polystyrene cylinder that can undergo strong bending deformations. The 3D data was captured using a MoCap system by tracking 30 mark-

ers. Figure 3.5 shows a few images of the object (with the markers) deforming. The 3D points were then projected onto an image sequence 676 frames long using an orthographic camera model. We evaluated the performance of the algorithm with respect to noise in the image measurements. Zero mean additive Gaussian noise was applied with standard deviation $\sigma = \mathrm{n} \times \mathrm{s}/100$ where n is the noise percentage and s is defined as the diameter of the **Woggle** in pixels. Noise levels of up to 30% were added. Figure 3.5 (right), shows the plot comparing the results obtained with our algorithm with those achieved using EM-LDS and BA-Lin. The plot depicts the 3D error averaged over 50 random runs after removing the results from tests that failed to converge showing an improved performance of the Quad algorithm versus EM-LDS and BA-Lin.

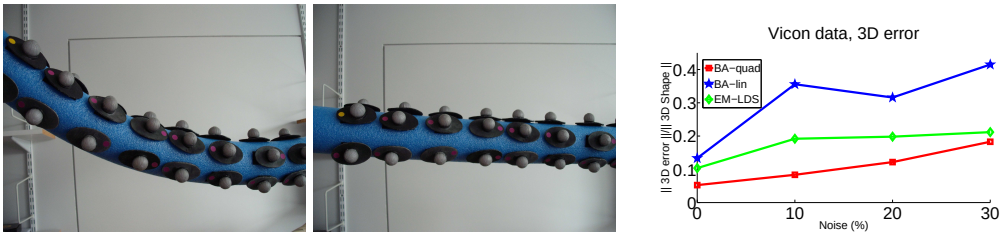


Figure 3.5: Left: Images of the **Woggle** used in the experiments with motion capture data. Right: Average 3D error plot for the reconstruction of the **Woggle** motion capture sequence , using only experiments that converged to a valid solution, and with increasing levels of noise. Our method (Quad) outperforms EM-LDS and BA-Lin as it provides lower 3D error even at the highest level of added noise tested.

Figure 3.6 shows the ground truth (green circles) and reconstructed 3D shapes (black dots) for five frames of the sequence in the absence of (added) noise using the three different algorithms. In the case where no noise was added to the motion capture data, our method recorded a reconstruction error of 5.25%, which is lower than the reconstruction errors of EM-LDS (9.37%) and BA-Lin (16.69%).

3.4.3 Real experiments

In Figure 3.7 we show the reconstruction of the **Cushion** sequence, in which 90 points were tracked during bending and stretching motion. For this sequence we compared

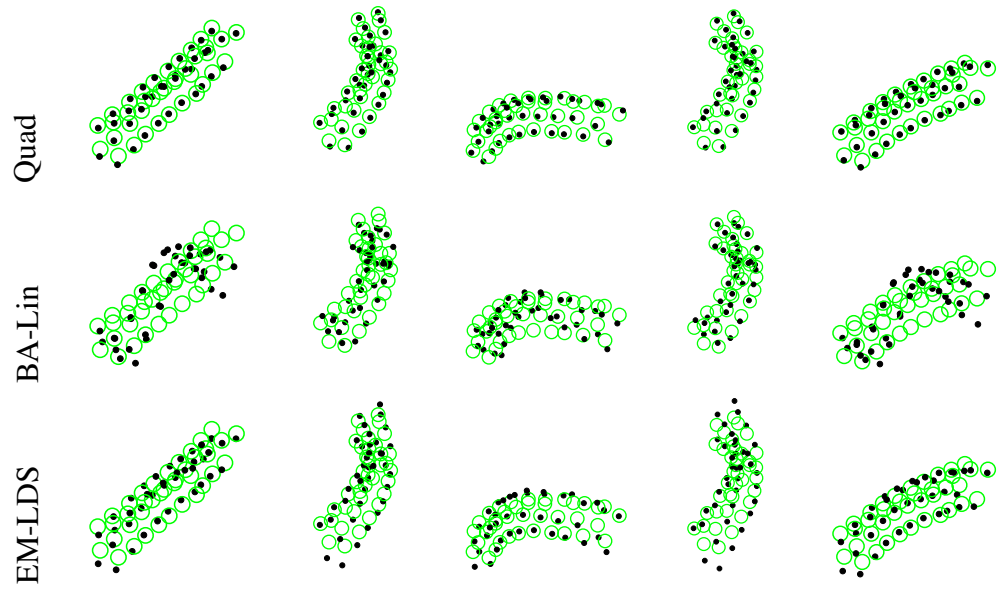


Figure 3.6: Example frames of the 3D reconstructions of the real **Woggle** MoCap data sequence obtained with the 3 different algorithms. The green circles correspond to the ground truth positions while the black dots correspond to the algorithm reconstructions.

the reconstruction of our Quad method with Torresani *et al.*'s EM-LDS algorithm. For a better visualization a mesh was then fit to the reconstructions with texture added from the first image where the cushion is facing the camera. While both methods have a reasonable frontal view reconstruction, which essentially shows that these methods are minimizing the re-projection error, when comparing the reconstruction on the side view it is clear that our method provides a more plausible reconstruction.

3.5 Conclusions

We proposed the QD model as a new way to describe non-rigid deformations, and we showed how that model can be used within a NRSfM formulation. We discussed the different deformation modes allowed by this model and how these modes can be easily disabled when prior information on the objects is available. Our proposed NRSfM method uses a non-linear optimisation scheme to minimize the image re-projection

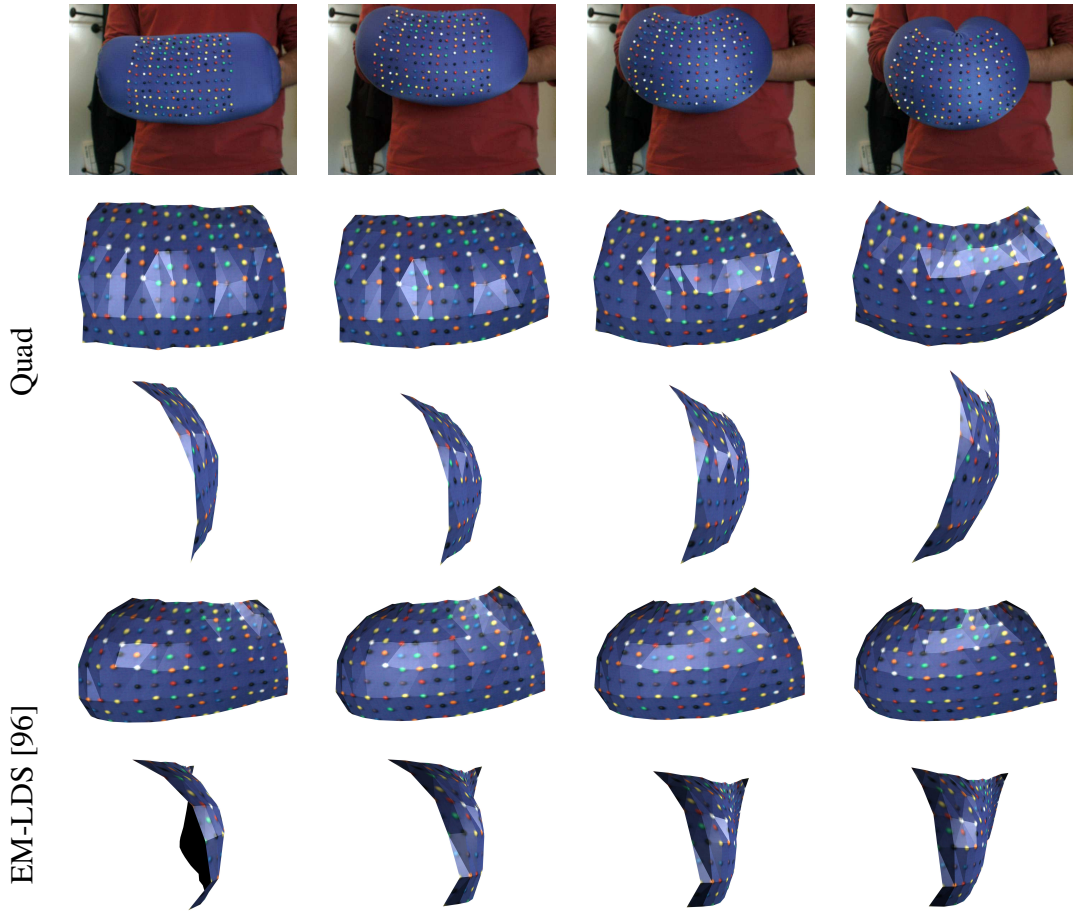


Figure 3.7: Top: Selected frames from the **Cushion** sequence, with bending and stretching motions. Rows 2 and 3: Front and top views of the 3D shapes for the selected frames using our new QD model. Rows 4 and 5: 3D reconstructions using EM-LDS

error given by modelling non-rigid motion with the QD model. As the Levenberg-Marquardt algorithm does not guarantee convergence to the global optimum, we proposed a good choice of initialization and show how the inclusion of temporal smoothness priors can help to constrain our solution and lower the chances of falling into local minima.

We presented two sets of quantitative tests, one using synthetically generated data and another one using real object deformations captured using motion capture technology. In our synthetic experiments we showed how the accuracy of our method compares to existing solutions using the low-rank shape basis model (Torresani *et al.*'s [96]

and a Bundle Adjustment algorithm [25]). Our results show that our proposed method has both a higher convergence rate, in terms of a statistical analysis of the 3D reconstruction error of sequences with a similar degree of deformation, and a lower average 3D reconstruction error for these sequences (see Figure 3.4). On the motion capture sequence, we used different levels of zero mean additive Gaussian noise to show the robustness of our method. As Figure 3.5 shows, our method has the lowest 3D reconstruction error for all the noise levels tested.

Our NRSfM algorithm is formulated assuming an orthographic camera model for the image acquisition. Such model relies on the the relative depths between the reconstructed points to be small when compared with the distance from the camera to the object. In our experiments we have not encountered any reconstruction errors that would justify using the more complex perspective model. This would increase the complexity of our cost function and possibly originate ambiguities between perspective effects and object deformations (e.g. a stretching motion could be confused with a translation of the object towards the camera). However there is no technical limitation preventing the usage of the perspective camera model. As a basic setup, the initialization could remain based on the orthographic camera model while the perspective camera model would be used in the non-linear optimization as a refinement. The study of the implications of using a perspective camera model are then left as future work.

There are two important limitations of our proposed approach that should be mentioned. To begin with, we have presented experiments that are characterized by strong bending and stretching motions, which are deformation modes present in our model. These experiments were conducted to show examples of deformations where our non-linear deformation modes would be preferable to the low-rank shape basis method, which would probably need a high number of shape basis to cope with such deformations, leading to overfitting and finally to a degradation of performance. However, we are aware that our method is only suitable when the object deforms globally in a combination of the deformation modes of the QD model. It cannot be expected that more

complex motions can be modelled with this approach. The second important limitation is the need for the object to undergo rigid motion in the beginning of the sequence to allow the recovery of the rest shape. Ideally there should be a way to recover it automatically from a sequence without constraining the possible motions. In order to tackle these two problems, we further developed our NRSfM approach by applying the QD model in a piece-wise approach. This work will be described in detail in Chapter 4.

Chapter 4

Piecewise Non-Rigid Structure from Motion with the Quadratic Deformation Model

As discussed in Chapter 2, piecewise approaches for non-rigid shape estimation are a recent trend in the NRSfM community [100, 90, 34, 22]. These approaches are based on the intuition that while the global motion of strongly deforming objects might have high dimensionality, local motion is more constrained and simpler to model. Local models require fewer parameters than global ones, and as each model is fitted to fewer points, they are both easier to optimise and are less prone to over-fitting. Given an independent solution to each patch, spatial consistency can then be enforced between these overlapping 3D patches to create a continuous global surface. Various different local models have been used in the literature including planar [100, 22] and rigid triangles [90] (see Section 2.2.3).

In this chapter we tackle the limitations of global models by arguing that local modelling of the deformations can achieve accurate reconstructions. Although our proposed method is general and applicable to a wide range of sequences, we focus on sequences where objects undergo strong deformations which makes the reconstruction

problem harder for global methods. As a practical example, let us consider the motion capture sequence acquired by White *et al*. [102] consisting of a flag waving in the wind where 450 points are captured during 540 frames. Some examples of the deformation which the flag undergoes can be seen in Figure 4.1, where texture was added to the point cloud to make the motion clearer. An object with such complex deformations cannot be modelled by the method proposed in Chapter 3, as it does not behave globally as a quadratic surface. Furthermore, when trying to reconstruct such motion with Torresani *et al*.'s [96] EM-LDS algorithm or with a Bundle Adjustment optimisation approach of the low-rank basis shape model [26] these methods also fail and the 3D error is very high (see Figure 4.7).

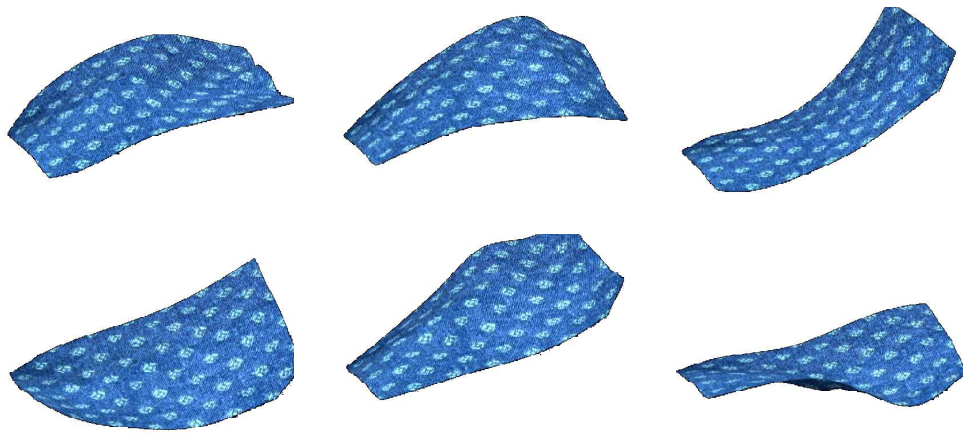


Figure 4.1: Some frames of the MoCap **Flag** [102] sequence with added texture for better visualisation.

Our proposed algorithm divides the surface into patches, reconstructs each of these patches individually and finally registers all the patches together enforcing global consistency to give a single smooth surface. Our method is generic in the sense that it does not rely on any specific reconstruction method for the individual patches. In principle, any SfM method can be applied locally. However, in our experiments, we have found that the Quadratic Deformation (QD) model provides the best local reconstructions (see Section 4.5). Additionally, as discussed in Chapter 3, the QD model has already been

used in three dimensional piecewise fashion in the field of computer graphics to increase the range of deformations it could model [66, 71]. We tackle the problem of rest-shape estimation (see Section 3.3.2) by considering three different scenarios, and providing a different approach for each case. We show results on challenging motion capture and real video sequences with strong deformations and a very small amount of camera rotation (which adds to the difficulty of obtaining accurate reconstructions) and where we show that global methods fail to provide good results.

4.1 Piecewise Non-Rigid Structure from Motion

The piecewise approach we propose draws on the intuition that modelling deformable objects globally is a very hard and ambiguous problem, while reconstructing local regions of objects independently is a more constrained problem. This is due to fact that the deformations these local regions can undergo will be less complex when compared to the global object deformations. Additionally, as described in Chapter 3, the deformation modes of the QD model have a clear physical meaning, such as stretching, shearing, bending or twisting. We argue that while these deformations cannot be expected to model deformable objects globally, they seem naturally suited to model deformations of local regions.

We formulate this problem in the same context as the NRSfM method described in Chapter 3, where an object with P points is observed across F frames with an orthographic camera. Since our goal is to reconstruct local regions independently, our first step should be to divide the object into local regions (or patches). These regions are in practice just a subset of the P points that belong to the object. Thus these local regions can be reconstructed by treating them as independent NRSfM problems. To reconstruct the original object these local reconstructions must be later merged into a single 3D point cloud. A simple way to perform this without needing extra constraints in the independent NRSfM problems is to make sure the patches overlap with each

other *i.e*. patches should share points with other patches. In this setup, a given point j that belongs to more than one patch will have more than one 3D reconstruction. However as they are all reconstructions of the same physical point, they should ideally all have the same 3D coordinates. This simple constraint turns out to be enough to combine all the local reconstructions into a single 3D point cloud. Our algorithm can be summarised as follows:

Require: 2D correspondences of points tracked through the sequence.

Ensure: 3D reconstruction of the global surface for every frame.

1: Divide surface into N regular patches $\mathcal{P} = \{\mathcal{P}_1, \cdots, \mathcal{P}_N\}$

2: Reconstruct individual patches using the QD model.

3: Combine individual 3D reconstructions.

4: Final optimisation.

Algorithm 1: Piecewise Reconstruction of Highly Deformable Surfaces

4.2 Shape Matrix Estimation and Division of the Object into Patches.

The aim of our piecewise approach is to provide a fully automatic method to deal with any type of 3D non-rigid surfaces, whether planar, such as a piece of paper, or non planar such as a beating heart or a torso. As we saw in Chapter 3, the NRSfM algorithm with the QD model can be formulated as a non-linear optimisation problem of minimizing the re-projection error:

$$\mathcal{R}(\mathbf{w}_{ij}, \mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i, \mathbf{s}_j) = ||\mathbf{w}_{ij} - \hat{\mathbf{w}}_{ij}||^2 = ||\mathbf{w}_{ij} - \mathtt{R}_i(\mathbf{q}_i)\,[\mathtt{L}_i\,\mathtt{Q}_i\,\mathtt{C}_i]\,\mathbf{s}_j - \mathbf{t}_i||^2\,, \tag{3.12}$$

together with temporal smoothness prior terms (see Equation 3.13 and Equation 3.14). As Equation 3.12 shows, the shape matrix $\mathtt{S}_q$ is not part of the parameters of the cost-

function, and thus it is not optimised. Recall that the shape matrix $\mathtt{S}_q$ is defined as:

$$\mathtt{S}_q = \left[\begin{array}{cccc} X_1 & X_2 & \dots & X_p \\ Y_1 & Y_2 & \dots & Y_p \\ Z_1 & Z_2 & \dots & Z_p \\ \hline X_1^2 & X_2^2 & \dots & X_p^2 \\ Y_1^2 & Y_2^2 & \dots & Y_p^2 \\ Z_1^2 & Z_2^2 & \dots & Z_p^2 \\ \hline X_1Y_1 & X_2Y_2 & \dots & X_pY_p \\ Y_1Z_1 & Y_2Z_2 & \dots & Y_pZ_p \\ Z_1X_1 & Z_2X_2 & \dots & Z_pX_p \end{array}\right] = \left[\begin{array}{c} \mathtt{S}^{(\mathtt{L})} \\ \hline \mathtt{S}^{(\mathtt{Q})} \\ \hline \mathtt{S}^{(\mathtt{C})} \end{array}\right], \tag{3.1}$$

which means it is fully specified by $\mathtt{S}^{(\mathtt{L})}$, the linear shape matrix (or *rest-shape* since it represents the shape of the object when no deformation coefficients are active). In Chapter 3 we recovered the rest-shape by assuming the object does not deform for the first few frames of the sequence, which was one of the limitations of our approach. In this section we show how we can relax this constraint under some circumstances. When performing NRSfM, often some *a priori* information exists about the nature of the object being observed. We thus analyse the object properties and provide a solution to the division of the surface into regular patches in three different situations: when a reference 3D shape or *template* is known for a reference image in the sequence, when the surface is known to be a planar shape but a 3D template is not available, and finally in the general case where no *a priori* knowledge is available about the surface. In every case, patches are obtained by dividing the object into a set of regular overlapping regions.

When dividing the object into patches, care must be taken so that each patch satisfies the reconstructibility requirements of the local NRSfM model chosen. In the case of the QD model, in order to initialize the patch assuming rigid motion in the first few

frames (see Section 3.3.2) the object must have at least 4 non-coplanar points. However the QD model adds a few more parameters per frame, resulting in an increase in the minimum number of points. Since after the estimation $\mathtt{S}$ it is kept fixed on the optimisation step, we can reduce our analysis to the reconstruction of a single frame. In Section 3.2 we have described some constraints applied on the 3×9 matrix $\mathtt{A}_i$ that reduce the number of coefficients to estimate from 27 to 21. Additionally, we must also estimate 3 parameters for each rotation matrix $\mathtt{R}_i$ and 2 parameters for each translation $\mathbf{t}_i$, giving a total of 26 parameters to estimate. Every point contributes with 2 additional equations per frame to the problem. Hence, this algorithm requires a minimum of 13 points to estimate all the deformable motion parameters per frame.

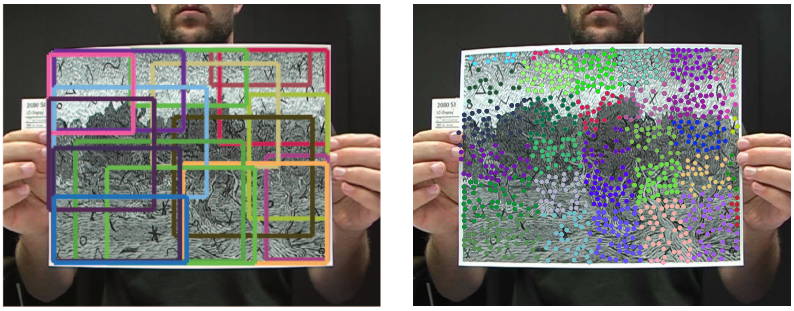
In practice, good quality reconstructions depend not only on fulfilling the minimum mathematical constraints of the problem, but also in assuring that patches will present motions consistent with the deformation modes of the model. For instance, if the minimum number of points for reconstruction with the QD model is fulfilled, but these points are located very near to each other on the object's surface, it is very likely that their motion will be quasi-rigid, failing to take advantage of the power of the QD model. As a rule of thumb for reasonable reconstruction, the size of the patches should not be chosen based on the number of points (provided the minimum number constraints are fulfilled), but on the area of object surface those points represent, and how likely it is for that area to be well explained by the QD model.

Additional care must be taken when choosing the width of overlap between patches. If the width is of one point, the constraint of overlapping points having the same 3D coordinates would be fulfilled by every reasonable reconstruction that kept the structure of the object intact. To guarantee second order smoothness over the patches, a width of at least two points in the overlap is required. In practice, the size of the overlap again depends more on the real object overlap area than on the number of points in the width of the object.

4.2.1 Known reference shape

As described in Chapter 2, there are a number of approaches to non-rigid shape reconstruction from monocular sequences that rely on the assumption that the shape of the object is known in some reference image [83, 82, 72, 21]. For instance, often the surfaces of interest are sheets of paper or cloth and it is reasonable to assume that they are viewed in a planar configuration in the first frame. If this assumption is satisfied, the rest shape is simply the planar configuration of the planar object, with the Z coordinate of $\mathtt{S}^{(\mathtt{L})}$ (and corresponding entries in $\mathtt{S}^{(\mathtt{Q})}$ and $\mathtt{S}^{(\mathtt{C})}$) being zero.

In such a case the object is divided into regular patches by specifying the number of intervals along the X and Y dimensions, and a percentage of overlap on every side of the patch. The division is done by creating a regular grid on top of the planar rest-shape and enlarging each region by the specified percentage of its size in all four directions. An example of such a division can be seen in Figure 4.2 where each rectangle represents the area of the image considered as a patch, and one can clearly see the overlapping regions amongst them.



(a) (b)

Figure 4.2: (a) Regular division into overlapping regions. Each different rectangle shows the area from which the patches will be constructed. rectangles have different sizes because they were cropped at the object boundary. (b) Patches in terms of point tracks. Different Colours represent different patches. Overlapping regions cannot be visualised as points and are only plotted with the colour of one of the multiple patches they "belong" to.

4.2.2 Planar surfaces

In some situations, we know in advance that the surface being reconstructed is a deforming plane (a sheet of paper or a flag waving in the wind), but a reference image for that shape is not known. In this case, we propose a method based on the isometric low-dimensional mapping method Isomap [91]. First we reconstruct a *mean* shape of the surface *i.e*. the shape that minimizes the re-projection error of the non-rigid sequence with rigid motion only, by applying Tomasi and Kanade's rigid factorization algorithm [93] to a few frames or to the entire sequence. Since the object is non-rigid, this average rigid surface will not be planar. Therefore it is not straight forward to divide it into regular patches. However, we can use Isomap [91] to compute an isometric low-dimensional embedding (the 2D flat surface) of the higher dimensional data (the *mean* 3D surface). In other words, Isomap will find an isometric mapping of the deformed *mean* surface, obtained by rigid factorization, onto a 2D plane. Figure 4.4(a)-(c) illustrates the process. Due to noise in the data and to the sparseness of the 2D tracks the embedding will not be exactly isometric. However, it is a good enough representation to use for the division of the surface into regular overlapping pieces.

It might be argued that instead of using the more complex Isomap to estimate the 2D embedding, it would be simpler to project the 3D mean shape to 2D or even to perform a rank-2 factorization and recover a planar shape. However these approaches do not attempt to preserve the true distance between the feature points. In sequences such as the **Paper**, where the object has a strong deformation along the Z axis, these distances will be shortened to a great extent. This would imply that patches defined over such shortened regions would require the QD model to 'rectify' such distortions with the deformation coefficients, which could cause problems in the reconstruction. As we have previously discussed in Section 3.3.2, the QD model relies on a reasonable initialization for the rest-shape. Thus, we prefer to use the more complex Isomap [91] approach, which will better preserve the 3D distances recovered from the 3D shape

obtained by rigid factorization.

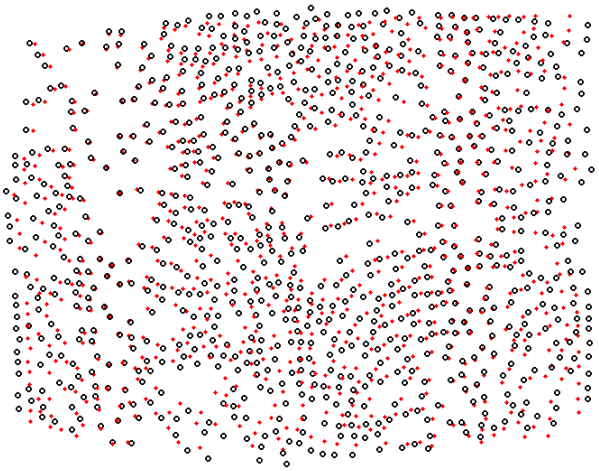
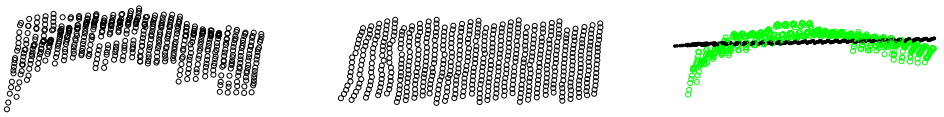


Figure 4.3: Comparison between the rest-shape estimated by Isomap [91] (red) and the rest-shape estimated by projecting the 3D shape recovered by factorization [64] (black) for the **Paper** sequence. Note how Isomap [91] better preserves the right angles of the paper.

Since the object is now planar, we have reverted back to the case of Section 4.2.1 and so we apply the same method do divide the object into regular patches.



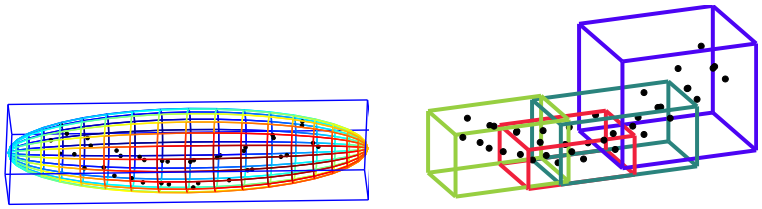
(a) (b) (c)

Figure 4.4: (a) Reconstructed mean shape of the **Flag** sequence (see Figure 4.8) using rigid factorization. (b) Result of applying Isomap to the surface. (c) Side view of the shape before and after Isomap.

4.2.3 Generic surfaces

If we know the object performs a rigid motion for the first few frames of the sequence we can apply the rigid factorization algorithm [93] to those frames to obtain a rest shape. If such knowledge is not available we can in turn perform rigid factorization over the whole sequence to obtain a *mean* shape. The regular division must now be into

regular volumes, and not regular planar regions. An ellipsoid is fitted to the rest-shape in order to estimate the volume of the object. Finally a bounding box of that volume is computed and divided into regular overlapping pieces. Figure 4.5 shows this process being applied to the Woggle sequence presented in Section 3.4.2.



(a) (b)

Figure 4.5: (a) Volume bounding box. (b) Division of volume into pieces.

4.3 Reconstruction of Individual Patches

Once the surface has been divided into a set of regular patches, each of these becomes an independent NRSfM problem. We highlight once again that the overall piecewise approach for NRSfM that we design does not imply the use of the QD model presented in Chapter 3 to solve these independent problems. Still, our intuition is that the QD model can encode bending, stretching, shearing and twisting modes of deformation which are natural ways in which objects deform locally.

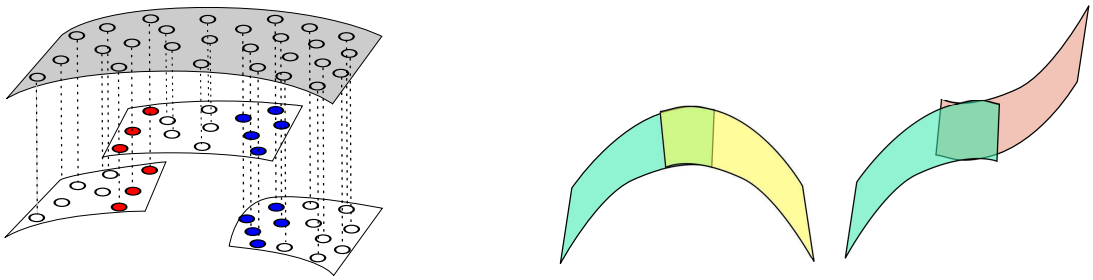
4.4 From Local Patches to a Global Reconstruction

The algorithm described in the previous section allows us to reconstruct the set of 3D patches $\mathcal{P} = \{\mathcal{P}_1, \cdots, \mathcal{P}_N\}$ independently. After solving the set of NRSfM problems we are left with the problem of combining them into a single object. As mentioned in Section 4.1, by dividing the patches into overlapping regions we can now use the

constraint that these regions of overlap (and corresponding points) are in fact the same 3D surface and must have the same 3D coordinates.

4.4.1 Resolving ambiguities: patch alignment

When performing reconstruction assuming an orthographic camera there are two ambiguities that cannot be resolved. To begin with, it is not possible to recover an absolute value for the translation along the camera viewing axis (the Z axis) as any translation along that axis results in the same 2D projection. In addition, there is an ambiguity regarding a concave or convex reconstruction of a given set of 2D tracks. If we imagine a solution $\hat{\mathtt{X}}$ that resulted from a given NRSfM method, due to the properties of the orthographic projection matrix, if at any instant i we replace the Z coordinate of $\hat{\mathtt{X}}_i$ by its symmetric value, the resulting cost will still be the same. Since our set of patches is reconstructed independently, each in their own reference frame, these reconstructions will not necessarily agree with each other (see Figure 4.6).



(a) (b)

Figure 4.6: (a) Reconstruction of shared points in different patches differ by a translation on the Z axis. (b) Representation of the ambiguity on the sign of the Z coordinate of the reconstructions.

While solving for the relative translation ambiguity is trivial given overlapping regions, correctly recovering the set of $N-1$ relative flips is an NP-hard problem (note that there is always a global flip ambiguity that cannot be recovered from, which is equivalent to fixing the flip of a given patch as a reference). We note that this ambiguity affects every frame independently. However, we rely on our smoothness terms

to impose flip consistency on a single patch, leaving only the global flip ambiguity to resolve. However, temporal smoothness cannot guarantee to resolve all the local ambiguities. For instance, in the case where a patch becomes fronto-parallel to the camera during the sequence the algorithm cannot distinguish between a concave or a convex deformation from that point onwards. In other words, every time a patch becomes fronto-parallel there will be a segment of the sequence for which a different flip ambiguity can arise. Our proposed approach only deals with a single global flip ambiguity and will in general fail if such more complex ambiguities arise.

To solve the relative flip and translation ambiguities we propose a greedy heuristic algorithm. Without loss of generality let us consider we have only two 3D surface patches to be aligned over the whole sequence, here named patch A and patch B. The alignment is done focusing on the P_{AB} points lying on the overlap of both patches. Each candidate 3D reconstruction of those points is represented by the $3\times P_{AB}$ matrices $\hat{\mathtt{X}}^{(A)}$ and $\hat{\mathtt{X}}^{(B)}$. As discussed in Section 4.2 we assume there is always a sufficient number of overlapping points that allow disambiguation. Since the image coordinates X and Y of every point are optimised by our formulation, only the Z axis will be altered in this process.

To solve the ambiguities we treat every candidate reconstruction as equally valid. We start the disambiguation process by registering the centroid of the overlapping areas for every frame. Once in this configuration, the choice of reflection ambiguity parameter can be formulated as follows:

$$\underset{\mathtt{X}=\left\{\hat{\mathtt{X}}^{(B)},\mathtt{Z}\hat{\mathtt{X}}^{(B)}\right\}}{\arg\min} \left|\left|\hat{\mathtt{X}}^{(A)}-\mathtt{X}\right|\right|^2, \mathtt{Z}=\begin{bmatrix}1&0&0\\0&1&0\\0&0&-1\end{bmatrix}, \tag{4.1}$$

which essentially minimizes the 3D distance between the shared points, after their centroids have been aligned at $Z=0$. After this ambiguity is resolved, we translate

both $\hat{\mathtt{X}}^{(A)}$ and $\hat{\mathtt{X}}^{(B)}$ along the Z axis back to the original position of $\hat{\mathtt{X}}^{(A)}$. Overlapping regions between patches have multiple 3D candidate reconstructions of the same 2D point. However, a one to one match between 2D and 3D points is desired. Thus, after registration, we merge both candidate reconstructions for each overlapping point by averaging them. Non-overlapping points of patch A and patch B are transformed in the same way. However there is no need for merging coordinates these points have only one reconstruction.

Although solving for the ambiguities is NP-hard, we encountered no problem with our heuristic algorithm provided that enough overlap between patches exists. Given the quadratic nature of our patches, the 3D distance between the overlapping points proved to be good disambiguation criterion as the curvature of correctly and incorrectly aligned patches results in very different values for our error measure.

4.4.2 Final Optimization

Once individual patches are reconstructed and initially aligned, a final global optimisation step is used to refine the results. This refinement is achieved by imposing the constraint that shared points must have the same 3D coordinates. This is done by applying the original cost function defined in Equation 3.14 to all the patches and adding a prior term that penalises reconstructions in which the 3D coordinates of shared points between patches are distant:

$$\sum_{i,j}^{F,P} \sum_{n \in \Theta_j} \left|\left| \mathbf{w}_{ij}^{(n)} - \hat{\mathbf{w}}_{ij}^{(n)} \right|\right|^2 + \lambda_\Theta \sum_{k \in \Theta_j / \{n\}} \left|\left| \hat{\mathbf{X}}_{ij}^{(n)} - \hat{\mathbf{X}}_{ij}^{(k)} \right|\right|^2, \tag{4.2}$$

where $\mathbf{w}_{ij}^{(n)}$ are the 2D coordinates of point j in frame i in patch (n), Θ_j is the set of N patches that contain point j, and $\hat{\mathbf{X}}_{ij}^{(n)}$ are the 3D coordinates of point j in frame i reconstructed from patch (n) using the QD model described in Chapter 3. This problem is solved using the Levenberg-Marquardt non-linear least-squares algorithm.

One could argue that this new optimisation step is able to solve the whole piecewise problem from an initial estimate of the set of parameters, without having to solve the ambiguities referred to in Section 4.4.1. However, non-linear least-squares requires the initial parameters to lie close to the solution, otherwise it can become trapped in local minima. Therefore, this final step is only used as a refinement of previous estimations to avoid possible ambiguities.

4.5 Experiments

Our approach aims at reconstructing highly deformable sequences where NRSfM methods based on global shape models fail. To be able to provide quantitative results and to allow comparisons with other methods, we have chosen to use a challenging example of a motion capture (MoCap) sequence of a flag waving in the wind [102]. This sequence is particularly difficult as it contains strong, rapidly varying deformations appearing through the whole surface. We show some frames of the MoCap **Flag** sequence with added texture in Figure 4.1.

4.5.1 Local vs Global modelling

Our first set of experiments was designed to show that current NRSfM models based on global models fail to achieve good reconstructions on a sequence of an object undergoing strong, agile or complex deformations. In Figure 4.7 we show ground truth 3D data together with some examples of 3D reconstructions obtained using 4 different global SfM methods: 1) (Quad) original global formulation of the QD model as described in Chapter 3, 2) (BA-Lin) linear combination of basis shape model with Bundle Adjustment optimisation [26], 3) (EM-LDS) NRSfM method proposed by Torresani *et al*. [96] and 4) Metric Projections method [69]. Note that the apparent stripe-like structure of the points on the **Flag** is not due to our piecewise reconstruction. It is present

in the ground truth 3D data as a consequence of the regular way in which the markers were placed. Table 4.1 (right) shows the reconstruction error given by the different algorithms. These experiments reveal that state of the art NRSfM methods based on global models fail to reconstruct this highly deforming object.

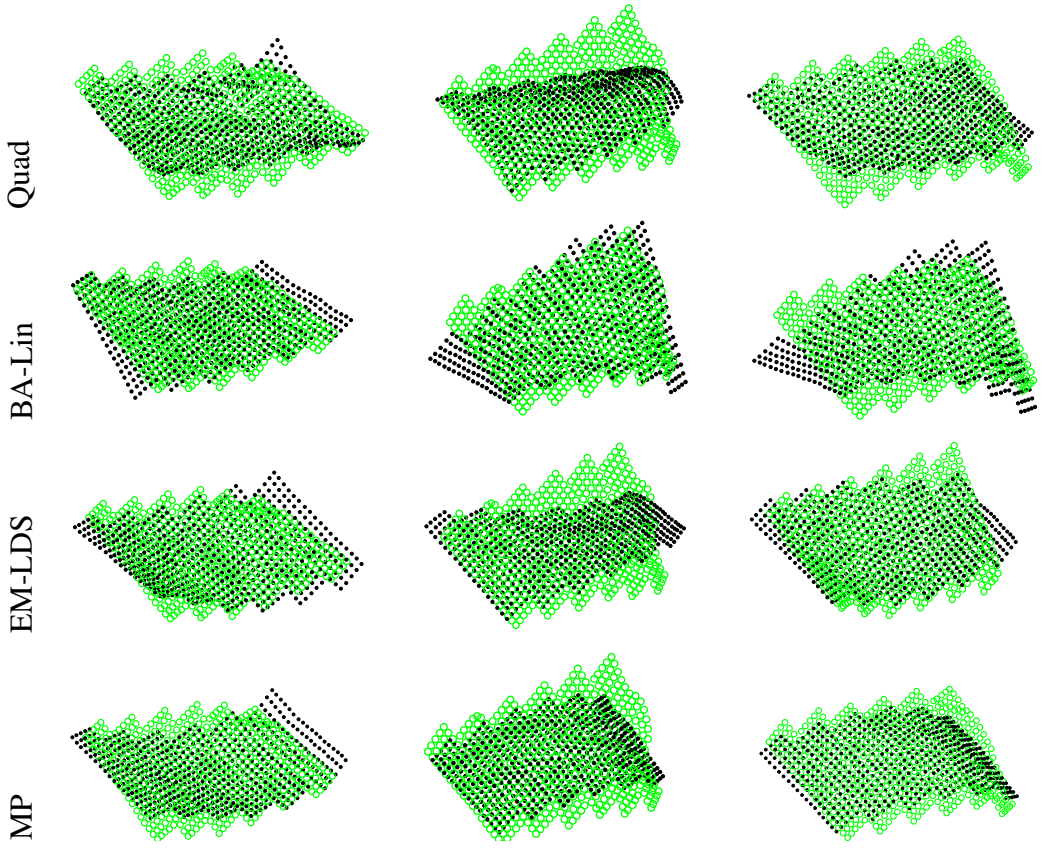


Figure 4.7: Reconstructions of the **Flag** sequence [102] using the Quad, BA-lin, EM-LDS and MP methods. Ground truth is represented by green circles while reconstructions are represented as black dots.

Justification of quadratic model as best local model

In this section we justify our choice of the QD model as the most adequate local model to express strong, natural local deformations. In Table 4.1 (middle column) we show the 3D reconstruction error (measured with respect to ground truth values) averaged over all the patches in the **Flag** for each of the algorithms mentioned in the previous

section. The 3D error is computed as described in Section 3.4.

It is clear from our results that the QD model outperforms all the other methods (4.05% error vs. errors between 15% and 29%). Each reconstruction algorithm was ran with its *out of the box* initialization. In the left column of Table 4.1 we show the average patch 3D errors when the mean shape for algorithms (BA-Lin) and (EM-LDS) and the rest shape for the (Quad) algorithm were initialized with the known ground truth flat shape given by the motion capture data. This experiment shows that *a priori* knowledge of the 3D shape of the surface improves the reconstructions. The quadratic model continues to outperform others by an order of magnitude (3.18% error vs. errors between 15% and 19%).

4.5.2 Piecewise quadratic reconstruction of MoCap sequences (flag and cylinder)

Applying the piecewise quadratic deformation model to the MoCap **Flag** sequence results in the reconstructions show in Figure 4.8 where the coloured points are the reconstructed points (colour encodes the patch they belong to) and the circles are the ground truth values. The rest shape was initialised from rigid factorization of 5 frames followed by flattening of the shape using Isomap. The object was divided into 36 overlapping patches.

Patch size ranges from 21 to 75 points, with an average size of 54.2 points, with the total number of points in the object being 540. A pair of overlapping patches share, on average, 17.6 points.

The 3D reconstruction error can be found in Table 4.1 (right column). Results show that in this challenging sequence, our model is able to provide a very accurate reconstruction, with only 3.25% of 3D error. Recall that the other NRSfM methods gave errors ranging between 15% and 26%.

In Figure 4.9 we show reconstructions (cyan dots) and ground truth values (black

circles) of the *MoCap cylinder* used in [36]. We report an average 3D error of 1.97% compared to a 3D error of 5.25% obtained in Chapter 3. Therefore the piecewise approach greatly improves the results of the global algorithm.

In this sequence the object was divided into 4 overlapping pieces, with two having 16 points and the other two 19 points, from a total of 39 points. A pair of overlapping pieces share, on average, 7.8 points.

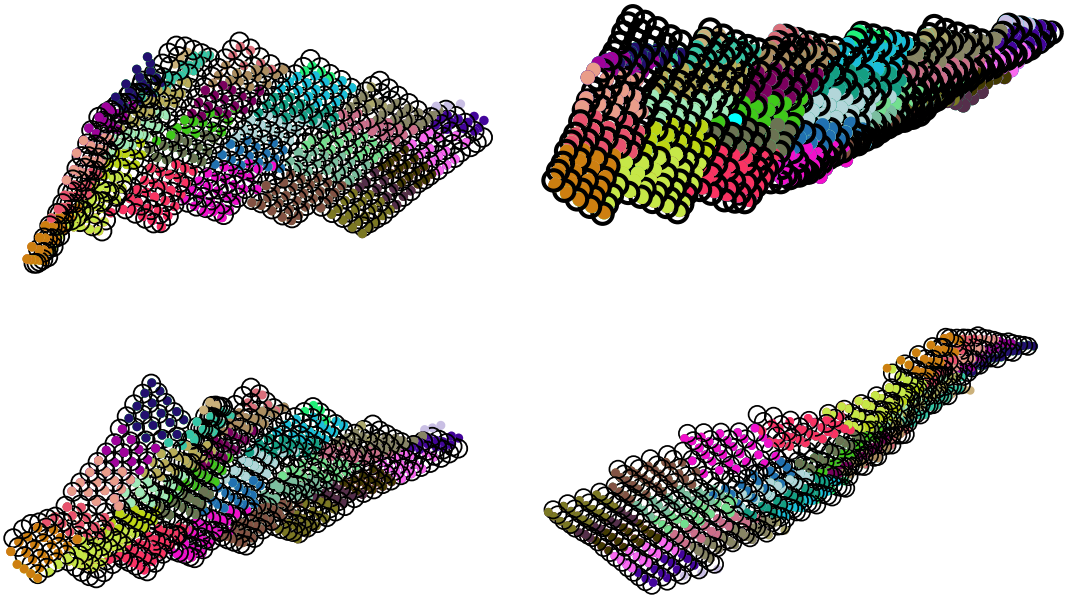


Figure 4.8: Reconstruction of 4 frames of the **Flag** sequence with our new piecewise quadratic deformations model. Ground truth is presented as black circles, reconstructed points are shown as coloured dots where the colour indicates the patch they belong to.

Table 4.1: 3D Reconstruction error for different NRSfM methods on the **Flag** sequence

Algorithm	**Patch GT init (%)**	**Patch Own init (%)**	**3D error whole Flag (%)**
Quad[36]	3.18	4.05	15.79
BA-Lin [26]	17.48	16.51	26.29
EM-LDS [96]	15.34	15.85	17.09
MP [69]	-	29.77	18.57
Piecewise-Quad	-	-	**3.25**

4.5.3 Piecewise quadratic reconstruction of real sequences (paper and back)

Figure 4.10 (top and middle rows) shows the reconstruction of the **Paper** sequence where a sheet of paper is bent [100]. Reconstructed points are represented in different colours representing the 36 patches used in the reconstruction.

In this case, the size of the patches ranges between 38 and 167 points, with an average size of 113 points, from a total of 871 points. A pair of overlapping patches share on average 31.47 points. The rest shape was obtained running rigid factorization on 8 frames and then using Isomap to obtain the 2D embedding plane. We also provide a qualitative comparison with the mesh obtained with Varol *et al.*'s method [100] (Figure 4.10,bottom row). When the deformation is strongest (last frame of the sequence) our reconstruction provides a more realistic curved shape, whereas Varol *et al.*'s appears to be a piecewise planar approximation. In addition, we present an example of augmented reality that illustrates the accuracy of our surface estimation. We show 5 pyramidal objects on top of the surface of the **Paper** that follow the bending motion. The re-projection of those objects over the original image fits appropriately, while the top vertex of each pyramid gives a notion of the surface normals at those points.

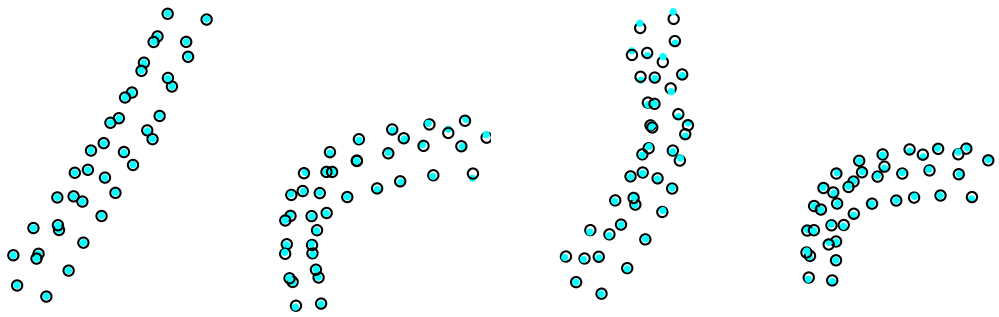


Figure 4.9: Results of the reconstruction of the *(MoCap cylinder)* sequence used in [36]. Blue dots are reconstructed points and black circles are ground truth values.

In addition, we evaluate our method on the **Back** sequence. This sequence shows

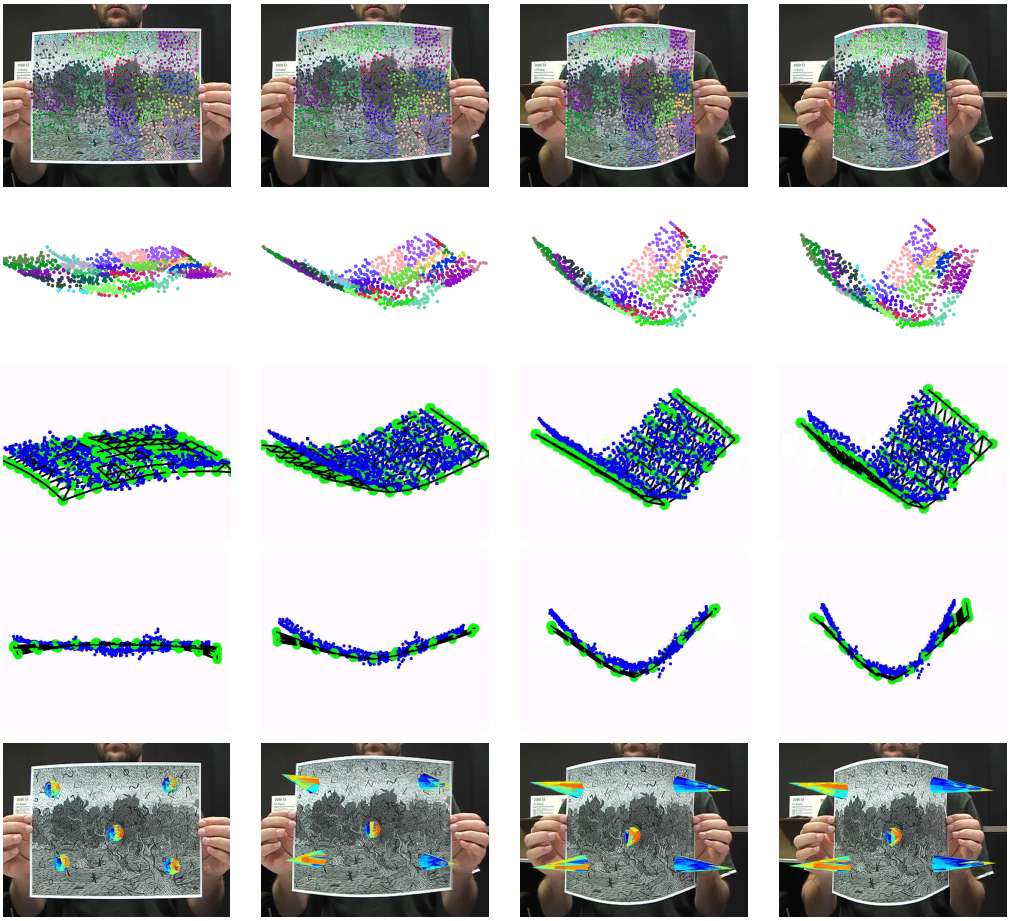


Figure 4.10: Reconstruction of **Paper** sequence [100]. The different colours show the different patches. First row: 2D re-projection of the points. Second row: 3D reconstructions with our piecewise reconstruction. Third and Fourth row: Comparison of our reconstruction (blue point cloud) with Varol *et al.*'s method [100] (mesh with green vertices). Fifth row: An example of augmented reality, where 5 pyramids are placed on top of the surface and their projected motion shown over the original image.

a man viewed from the back while he moves his torso to create natural non-rigid motion. This dataset comes from [85], where the coloured dots on the garment are meant to be reconstructed using stereo pairs. We use the 2D tracks provided by [85] as input and measure a reconstruction error of 15.2% when considering the stereo-based reconstruction as ground truth. Figure 4.11 illustrates our reconstructions.

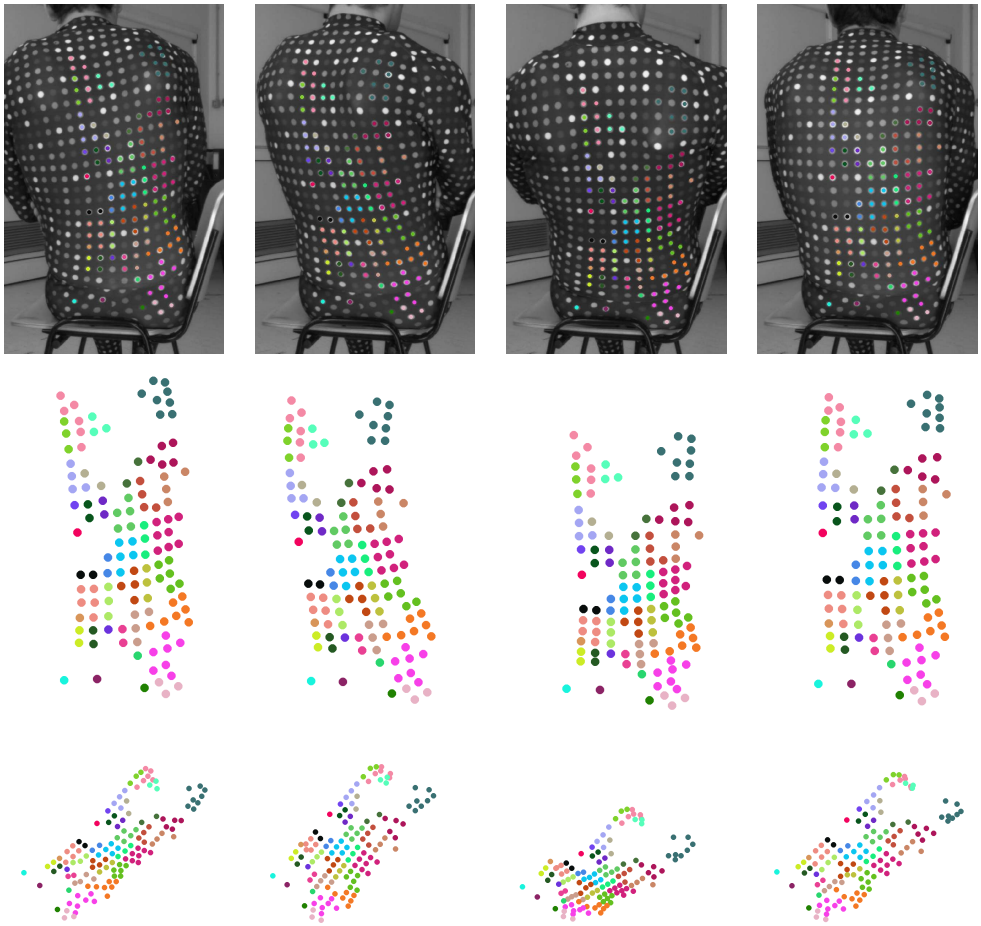


Figure 4.11: Reconstructions of the **Back** sequence from [85]. Top row shows the point correspondences. Different colours represent different patches. Middle and bottom rows show the results from our 3D reconstruction. The reconstruction error is 15.2%.

4.6 Conclusions

In this chapter we analysed how several state of the art NRSfM methods that model non-rigid objects globally behave when reconstructing sequences that are characterized by strong and agile deformations. We formulated a hypothesis that such deformations lead to overfitting when modelled globally and provided some experimental results to support our claim. Following this view, we proposed that such deformations are better modelled locally and thus proposed a piecewise approach for NRSfM which divides the object into overlapping patches, solves NRSfM problem of reconstructing each patch independently, and later stitches them back together into the 3D global

reconstruction of the non-rigid object. We provided experimental results to justify the Quad algorithm presented in Chapter 3 as our local NRSfM approach, although any other NRSfM method can be used in our piecewise formulation.

We provided quantitative experimental results by using motion capture sequences to measure the 3D reconstruction error. We compared with the state of the art global methods and showed that our Piecewise-Quad algorithm outperformed these global approaches, including the (global) Quad algorithm presented in Chapter 3. We also provided qualitative analysis in challenging real sequences and compared our results with the piecewise planar approach from Varol *et al*. [100], showing that the Quadratic Deformation Model has an advantage of providing smoother reconstructions for the kind of sequences such as the **Paper** sequence shown in Figure 4.10. A comparison of the algorithms presented so far can be found in Table 4.2

Table 4.2: Summary of presented algorithms.

Algorithm	**Piecewise**	**Model**	**Initialization**
Quad (Chapter 3)	No	QD	Rigid SfM (from first few frames)
Piecewise-Quad (Chapter 4)	Yes	QD	Rigid SfM (+ Isomap if known to be flat)

Throughout this chapter our piecewise formulation acted on a set of overlapping patches that were generated by manually controlled regular division. Although experimentally the reconstruction results with regular patches were good, it is clear that they depend on a good choice of patch division. Intuitively, it is easy to see how regions of different sizes and shapes might be a better fit to a generic non-rigid motion. In Chapter 5 we propose a principled approach to determine the number and shape of the patches without prior information on the deformable motion being observed.

Chapter 5

Networks of Overlapping models for Non-Rigid Structure from Motion

In Chapter 4 we saw how global methods for NRSfM have trouble reconstructing non-rigid motion with strong deformations in multiple local regions, as they require a substantial increase in the number of basis shapes used, which tends to cause over-fitting. This limitation of global methods pushed us to develop a piecewise approach for NRSfM where the key idea is to split the object to be reconstructed into overlapping regions, each of which is modelled independently. Local models require fewer parameters than global ones, and as each model is fitted to fewer points, they are both easier to optimise and are less prone to over-fitting. Despite proving effective at reconstructing highly deformable surfaces, this piecewise method suffers from an important drawback. The problem of providing a principled formulation for the division of the surface into models was overlooked, with the patches chosen by dividing the object into regular overlapping patches.

In this chapter we formulate the problems of model assignment and model fitting as minimizing a geometric fitting cost, subject to a spatial constraint that neighbouring points should also belong to the same model. Under this formulation, we are able to jointly optimise the assignment of points to models, and the fitting of models to points,

to minimize this fitting cost. This gives a principled joint formulation for patch division and 3D reconstruction which results in an *adaptive* method where the size and shape of patches are optimized based on the observed 3D motion. This in turn leads to simpler 3D reconstructions with substantially lower 3D errors.

A fundamental requirement for piecewise reconstruction is the need for overlap between models to enforce global consistency, and to encourage smooth transitions between models. We capture this in our formulation by allowing feature points on the border between models to have more than one label or, equivalently, to belong to more than one model. Such overlaps are unsupported by current approaches that follow the Expectation-Maximization (EM) [30] paradigm such as PEARL [53], or K-means [61]. To meet this requirement, we will use the Networks of Overlapping Models formulation developed by Russell *et al*. [77], which allows for points that lie at the boundary of two models to have more than one label. This approach differs from standard soft assignment clusterings [58, 61], in that: *(i)* neighbours adjoining a point are encouraged to belong to the same models as this point; *(ii)* the sum of fractional assignments over a point need not add up to 1; and *(iii)* it incorporates a minimum description length (MDL) cost. This energy for fitting overlapping models can be optimised effectively with a simple hill climbing approach which makes use of a variant of the graph-cuts based algorithm α-expansion [77].

5.1 Graph-cuts Based Model Assignment

In these applications, rather than labels representing a fixed set of object classes or stereo disparities, the labels represent parameters of a model that must be fitted to the data. The parameters, and the assignment of points to an instance, are chosen to minimize some form of fitting error, and to respect spatial constraints which say that neighbouring points should normally belong to the same model, or that changes in labelling should vary smoothly.

Another problem arises in the fitting of models to their assigned points: The algorithm PEARL [53], uses an EM approach which alternates between assigning points to models, and fitting models to points (see also Algorithm 2). However, the presence of the previously discussed pairwise terms of [10] which penalize curvature between neighbouring points, means that fitting a model to its set of assigned points may increase the cost of an assignment. Consequently, an optimal choice of model to minimize the cost of assignment cannot be found. Because of this, in [10], re-fitted models must be treated as new models rather than as a correction of the original model, and this further increases the complexity of inference.

In our formulation we will use the approach by Russel *et al*. [77] which proposes a simple alternative to the use of such 'smoothing' terms between points belonging to separate models. Instead of relying on pairwise energy minimization terms to smooth disparities, and fitting disjoint models to separate patches, Russell*et al*. [77] propose a novel energy minimization framework which fits overlapping models. In this framework, these smoothness constraints between multiple models, which are difficult to optimise, are replaced with an analogous constraint that these models must explain some of the same data. The resulting cost function can be easily optimised.

5.1.1 Minimum Description Length (MDL) costs

The approaches we have discussed propose new models to explain different regions of the image, by selecting from the best set of proposals. Consequently, they are prone to over-fitting, and often propose near identical models for disjoint regions of an image that should share the same model. To overcome this, a penalty cost may be imposed, based on the number of models present in an image [10, 29, 50]. This model cost may be a monotonically increasing cost which is *linear* (proposed in [50] and used in [10]), *concave* [29] with optimal moves proposed by α-expansion, or an *arbitrary* monotone increasing with sub-optimal moves by α-expansion [57].

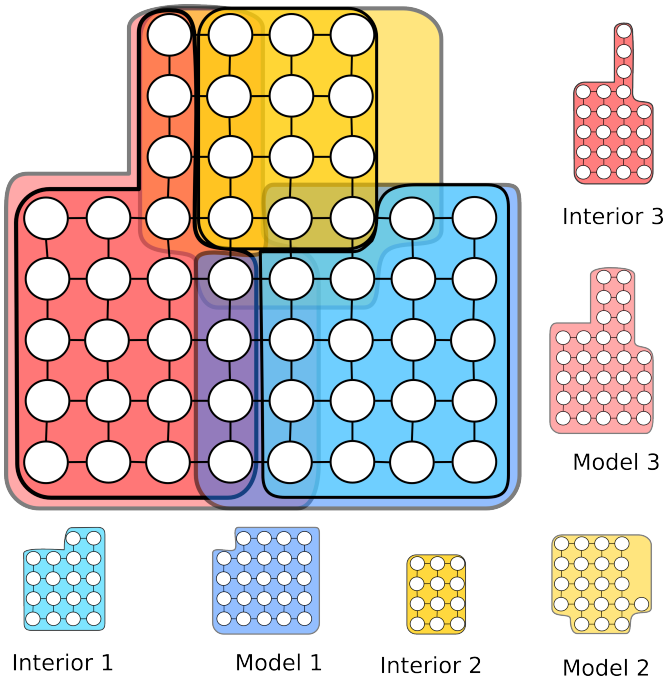


Figure 5.1: A simple grid structured graph, and a possible assignment of models, that satisfies constraints (5.2, 5.3). See Section 5.2 for more details. Best viewed in colour.

A significant contribution of these works, was in proving that these label set costs could be efficiently solved with α-expansion. We make use of this in Sections 5.2 and 5.3.1 by showing how the costs induced by overlapping patches can be reformulated as costs on the labels present in various neighbourhoods in the graph.

5.2 Formulating Multiple Model Assignment

To describe the problem of multiple model assignment, we require some notation: Given a set of points $\mathcal{P}$, for each $p \in \mathcal{P}$ we define a neighbourhood set $\mathcal{N}_p$ of adjacent points[1]. Assuming we have a set of models $\mathcal{M}$, we wish to assign a subset of these models m_p to each point $p \in \mathcal{P}$. This assignment should: *(i)* Cover the set $\mathcal{P}$. Every point $p \in \mathcal{P}$, should belong to at least one model, *i.e.* $m_p \neq \emptyset$. *(ii)* Adjacent models must overlap *i.e.* they must explain some of the same points. *(iii)* Minimize the accu-

[1]For notational convenience, we assume that each point belongs to its own neighbourhood.

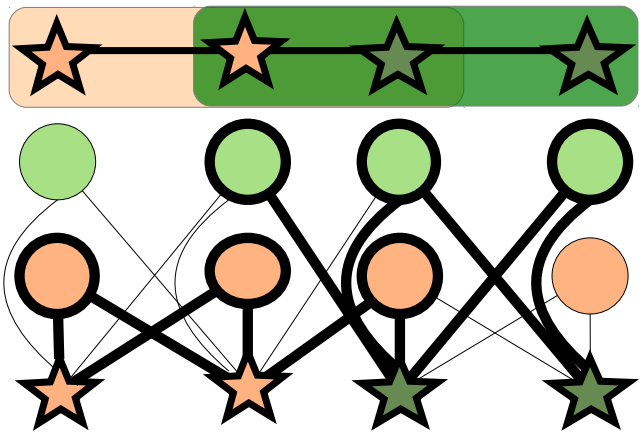


Figure 5.2: A transformation of the higher-order potentials into a pairwise graph. The top layer shows overlapping assignments of points (stars) to models (green and orange), below this can be seen the pairwise form of the cost function. In the pairwise form, we associate a single interior label with each point (again the stars), while the circles are auxiliary binary variables that indicate if a point belongs to a particular model. The strong black lines indicate active connections that currently force a binary variable to turn on, because one of the neighbouring points has an interior label that matches the model associated with a binary variable. To make the cost of the pairwise graph correspond to the higher-order costs of Equation 5.5 we give a binary variable that corresponds point p belonging to model α a cost of $U_p(\alpha)$ if it is turned on and 0 if it is turned off.

mulated error. This error is defined as the difference between the predicted 2D location of a point by its assigned models, and the observed position of the point (see Section 5.3.2 for details). As these terms are analogous to unary potentials, we will refer to the cost of fitting model α to point p as $U_p(\alpha)$. Note that $U_p(\alpha) \geq 0 \; \forall \alpha, p$. We will use $\mathbf{m}$ to refer to an assignment of a set of models to every point.

A naive formulation of the overlap constraint would simply say that all neighbours of a point must be assigned to the same models. However this constraint would propagate throughout the neighbourhood graph and force all connected components to be assigned to the same models.

Instead, we introduce the concept of interior points. As in topology, we define an interior point q of a model α as one whose neighbours p must also belong to the model

α, *but not necessarily as an interior point* [2]. Just as we use $\mathbf{m}$ to refer to the model assignment of every point, we use $\mathbf{I}$ to refer to the assignment of the interior of models to each point. Unlike our earlier formulation, the constraint that a point p belongs to the interior of model α does not force every other point to belong to α. However, as this supports a degenerate solution in which every p is not an interior point, and $m_p = \emptyset \, \forall p$, we enforce the constraint that every p must be an interior point of some model, or more formally, $\forall p \, \exists \alpha \, : \, p \in I_\alpha$. This constraint also guarantees that adjacent models must overlap. See Figure 5.1 for an example of a valid labelling of such models.

As the accumulated fitting error is simply the sum over all points and models of the unary term $U_p(\alpha)$, we can write down a cost function $C(\cdot)$ to minimize. We seek

$$\underset{\mathbf{m} \in (2^{\mathcal{M}})^{\mathcal{P}}}{\arg\min} \, C(\mathbf{m}) = \sum_{p \in \mathcal{P}} \left(\sum_{\alpha \in m_p} U_p(\alpha) \right), \tag{5.1}$$

subject to the constraints

$$\forall p \in \mathcal{P} \; \exists \alpha : p \in I_\alpha, \tag{5.2}$$

and

$$\forall q \in \mathcal{N}_p \wedge q \in I_\alpha \implies \alpha \in m_p. \tag{5.3}$$

Although well formulated, this problem is extremely challenging to optimise. Typically, the inference algorithms used in vision function under the assumption that exactly one model is fitted to a point, and this restriction gives a search space of size $|\mathcal{M}^{\mathcal{P}}|$ versus the $|(2^{\mathcal{M}})^{\mathcal{P}}|$ of our formulation. Moreover, the techniques used to efficiently solve large scale discrete problems such as α-expansion [15] or TRW-S [55] are designed to optimise pairwise cost functions over an unconstrained label space, and unable to optimise complex higher-order constraints such as (5.3) defined over large cliques. To make the above cost function tractable, we require two results:

[2] The mathematical formulation of this constraint is given in eq. (5.3).

Lemma 1. *A minimal cost solution* $\mathbf{m}$ *exists such that for all* p*, there exists a unique model* α *such that* $p \in I_\alpha$ *and* $p \notin I_\beta, \forall \beta \neq \alpha$*.*

Proof. By definition, every valid solution satisfies the constraint that $\exists \alpha : p \in I_\alpha$. Consider a valid solution of minimal cost, where $p \in I_\alpha$, $p \in I_\beta$ and $\alpha \neq \beta$. Removing p from I_β does not violate constraints (5.2) or (5.3) and does not increase the cost of (5.1), which only depends on $\mathbf{m}$. Ergo, it is also a valid minimal cost solution. As the set of points and models we consider is finite, by repeated application of this technique, we can arrive at a solution in which for all p, there exists a unique model α such that $p \in I_\alpha$ and $p \notin I_\beta, \forall \beta \neq \alpha$. □

Lemma 2. *If* $\mathbf{m}$ *is a minimal cost solution we can rewrite the cost* $C(\mathbf{m})$ *as*

$$C(\mathbf{m}) = \sum_{p \in \mathcal{P}} \left(\sum_{\bigcup_{q \in \mathcal{N}_p} \{\alpha : q \in I_\alpha\}} U_p(\alpha) \right). \tag{5.4}$$

Proof. As the error $U_p(\alpha) \geq 0$, following (5.3), a minimal cost solution occurs when m_p has as few elements in it as possible *i.e.* $m_p = \bigcup_{q \in \mathcal{N}_p} \{\alpha : q \in I_\alpha\}$ for all points p. This gives rise to the cost (5.4). □

Together, these two results suggest an optimisation strategy. We can eliminate the terms m_p from the equation and optimise over I_p in its reduced form, given in lemma 2. This results in an unconstrained cost function of the form

$$\underset{\mathbf{I} \in \mathcal{M}^\mathcal{P}}{\arg\min}\, C(\mathbf{I}) = \sum_{p \in \mathcal{P}} \left(\sum_{\bigcup_{q \in \mathcal{N}_p} \{\alpha : q \in I_\alpha\}} U_p(\alpha) \right). \tag{5.5}$$

Although this cost is higher-order, it is much closer to standard optimisation problems, and functions in a significantly reduced space. In fact, this cost function is equivalent to a unique label set cost defined over each neighbourhood, where a cost $U_p(\alpha)$ is added for every new label α introduced to a neighbourhood. We will make use of this

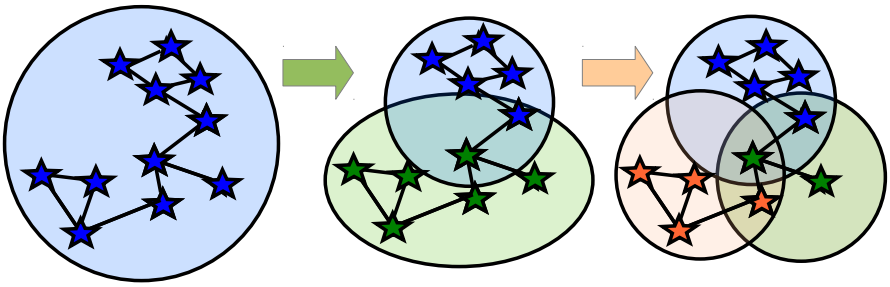


Figure 5.3: Evolution of the labels during α-expansion. The stars represent the points we want to label. The circles represent the models. First, all points are labelled with the blue model. Second, the green model is swept, changing the interior point assignment according to the cost, and creating the overlapping region. Finally the orange label is swept, ending the labelling process.

in showing that this reduced form energy can be optimised effectively using graph-cuts based α-expansion (see Figure 5.3).

5.2.1 Adjusting the Framework

Encouraging Overlap

In practice the cost function we have described penalizes overlapping regions too harshly for our purposes. Faced with a large region of overlapping models, our approach is likely to eliminate the overlap by removing one model entirely. To allow large areas of overlap to form, we use a variant on the cost function of (5.1),

$$C(\mathbf{I}, \mathbf{m}) = \sum_{p \in \mathcal{P}} \left(\lambda \sum_{\alpha \in m_p} U'_p(\alpha) + (1 - \lambda) \sum_{\alpha \in I_p} U'_p(\alpha) \right) \tag{5.6}$$

such that constraints (5.2, 5.3) hold, and $\lambda \in [0, 1]$.

A small value of λ down-weighs the cost paid by overlapping regions relative to the cost of assigning an interior point to a model, and allows for large regions of overlap to form. Note that the new term $(1 - \lambda) \sum_{\alpha \in I_p} U'_p(\alpha)$ can be seen as a unary potential defined over I_p, and its presence does not alter the reduction of the cost function

described in section 5.2, nor the inference in section 5.3.1. Following the derivation given in section 5.2, the reduced and unconstrained form of this weighted cost function is

$$C(\mathbf{I}) = \sum_{p \in \mathcal{P}} \left(\lambda \sum_{\bigcup_{q \in \mathcal{N}_p} \{\alpha : q \in I_\alpha\}} U'_p(\alpha) + (1 - \lambda) \sum_{\alpha \in I_p} U'_p(\alpha) \right). \tag{5.7}$$

Note that if $\lambda \neq 1$, the model fitting described in section 5.3.2 minimizes the *weighted* image re-projection error, where the interior points of a model have a weight of 1 and all other points a weight of λ. In all experiments, we uniformly set $\lambda = 0.1$.

Minimum Description Length (MDL) costs

As with the works discussed in section 5.1.1, we also wish to discourage over-fitting, and to encourage disconnected regions to share the same model where appropriate. This can be done with an MDL based cost over the set $\mathbf{m}$. Using the same arguments as in lemmas 1, 2 it can be shown that if the MDL cost is monotone increasing [57], this is equivalent to an MDL cost over the set $\mathbf{I}$, in a minimal cost labelling.

This gives the cost

$$C'(\mathbf{I}) = C(\mathbf{I}) + \text{MDL}(\mathbf{I}) \tag{5.8}$$

where C is defined in equation (5.7), and $\text{MDL}(\cdot)$ is an MDL cost as described in [29, 50, 57].

Robustness to Outliers and Unwanted Model Overlap

Outliers may be handled in multiple ways. In particular it is not clear if a point should be considered an outlier of just one model at a time, or of all models simultaneously. We choose to describe points as outliers with respect to particular models as this brings several advantages. Even though outlier classification is done per model, a point can still be an outlier of every model simultaneously. This allows us to recover from erroneous point tracks, which would not be explained by any of the QD model patches.

Most importantly, the ability to label points as outliers of individual models allows us to avoid model overlap between neighbouring models that have too different motion. For instance, it is possible that when building the neighbourhood structure some points of the background are connected to points in the object surface. When reconstructing such points with a model from the object, their reprojection error will be high. By thresholding high 2D reprojection errors, we can label the background points as outliers of object models, and the object points as outliers of the background model. In this way, these models will not overlap, but points will still be reconstructed by the models for which they are inliers.

In order to do this, we say that a point p, may belong to a model α with a cost of $U_p(\alpha)$ or it may belong to model α as an outlier, with a cost of $\lim$. In point assignment, this is equivalent to replacing the terms $U_p(\alpha)$ in equation (5.5) with

$$U'_p(\alpha) = \min(U_p(\alpha), \lim). \tag{5.9}$$

If a point belongs to a model as an outlier, we no longer fit the model to this point (see section 5.3.2), but only to the set of inliers associated with the model.

5.3 Simultaneous Point Assignment and Model Fitting

We wish to find an optimal assignment of points to models and an optimal choice of model parameters to explain their assigned points. Our proposed solution is in line with EM approach [30] as we repeatedly alternate between finding a better assignment of points which satisfies the constraints of section 5.2, with the fitting of models to their assigned points. As discussed, this differs from conventional EM approaches in that points are assigned to multiple models. The algorithm halts when the accumulated error no longer decreases (see Algorithm 2). We discuss the efficient assignment of points in section 5.3.1 and the fitting of the model to the points in section 5.3.2.

```
Δ = −1;
while (Δ < 0) do
    CurrentError = GetError();
    Points = BestAssignment(CurrentModels);
    CurrentModels = BestFit(Points);
    NewError = GetError();
    Δ = NewError−CurrentError;
end
```

Algorithm 2: Model Fitting following the EM paradigm [30].

5.3.1 Point Assignment

α-expansion functions by 'sweeping' out a model hypothesis α across a graph, potentially replacing the current interior model γ_p, at any point p, with some pre-chosen α. The best possible expansion move is chosen, and this process is repeated on the resulting labelling, with different choices of $\alpha \in \mathcal{M}$, until convergence (see Figure 5.3). To demonstrate that α-expansion over $\mathbf{I}$ can be efficiently computed, we show that computing the optimal expansion move can be formulated as the minimization of a pairwise sub-modular energy and consequently can be solved using graph-cuts.

Formulating the expansion costs as a pairwise energy requires us to restructure the higher-order cost of (5.4) as a pairwise cost via the introduction of auxiliary indicator variables. To do this we note that cost

$$\sum_{\bigcup_{q \in \mathcal{N}_p} \{\alpha : q \in I_\alpha\}} U_p(\alpha) \tag{5.10}$$

is an MDL or label-set cost on $\mathbf{I}$ within the neighbourhood N_p *i.e.* if we consider the cost (5.10) and the neighbourhood N_p in isolation, we pay a fixed cost of $U_p(\alpha)$ for the presence of a particular label α in that neighbourhood. As this cost is monotonically increasing and linear, optimal moves can be computed using the techniques of [29, 50]. As we must solve many of these overlapping problems simultaneously, we are unable to use the efficient move proposal technique of [29], which halves the number of edges required in the graph and instead use the standard construct shown in Figure 5.4.

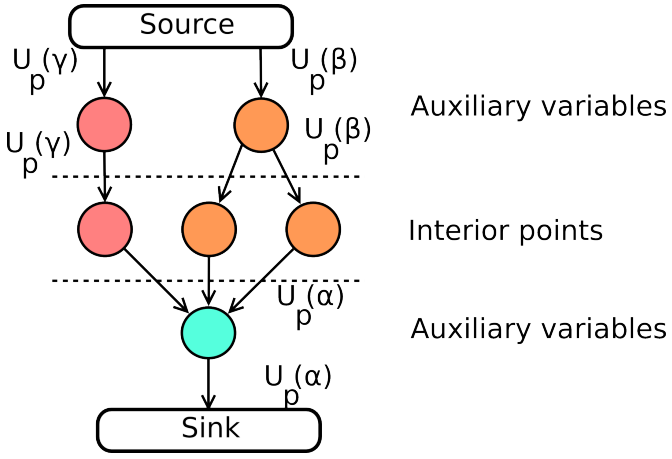


Figure 5.4: An example graph construct used to encode the costs of an α-expansion move in a single neighbourhood about point p, if one point in the neighbourhood currently takes label γ, and two take label β. Values of the form: $U_p(\alpha)$ indicate the capacity of edges. If the minimum cut leaves variables attached to the sink (bottom) these variables do not change label. If the minimum cut ties them to the source (top), they transition to take label α. See Section 5.3.1 for more details.

5.3.2 Fitting the model

Similarly to our piecewise approach from Chapter 4, one of the clear advantages of this new formulation is its independence from the model chosen to describe the deformations of individual patches. However in Chapter 4 we justified our choice for the QD model described in Chapter 3 as our local deformation model. Not only does this model have physically grounded deformations that seem intuitively quite suitable for local deformation modelling, but we also backed our claim with experimental results on challenging deformable motion.

After the point assignment step, each patch will then be reconstructed using the QD model based algorithm described in Chapter 3. In this formulation, the object is represented by an augmented shape matrix $\mathtt{S}_q$ containing a linear, quadratic and cross-terms shape matrices (see Equation 3.1). This matrix is entirely described by the choice of the Linear Shape matrix $\mathtt{S}^{(\mathtt{L})}$ which we have also named the rest-shape. This shape matrix $\mathtt{S}_q$ represents the set of points we want to label, as $\mathtt{S}_q$ is estimated once before the alternation and kept fixed. For more details on how to estimate $\mathtt{S}_q$ see Chapter 4.

The QD model is then characterized, at each image i, by a rotation matrix $\mathtt{R}_i$, a translation $\mathbf{t}_i$, and the three deformation coefficient matrices $\mathtt{L}_i$, $\mathtt{Q}_i$ and $\mathtt{C}_i$. These coefficients are fit to the points by minimizing the re-projection error

$$\mathcal{R}(\mathbf{w}_{ij}, \mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i, \mathbf{s}_j) = ||\mathbf{w}_{ij} - \hat{\mathbf{w}}_{ij}||^2 = ||\mathbf{w}_{ij} - \Pi\mathtt{R}_i(\mathbf{q}_i)\,[\mathtt{L}_i\,\mathtt{Q}_i\,\mathtt{C}_i]\,\mathbf{s}_j - \mathbf{t}_i||^2\,, \tag{3.12}$$

using the Levenberg-Marquardt non-linear least-squares algorithm (for more details see Chapter 3), where $\mathbf{w}_{ij}$ is the image position of point j at image i and $\mathbf{s}_j$ the j-th column of $\mathtt{S}_q$. As mentioned in Section 5.2, we will also use the re-projection error as our unary potentials in the labelling problem. To keep notations consistent, we can now formulate the cost of assigning point p to a specific model α with QD model parameters $\{\mathbf{q}_i^\alpha, \mathbf{t}_i^\alpha, \mathtt{L}_i^\alpha, \mathtt{Q}_i^\alpha, \mathtt{C}_i^\alpha\}\ \forall i \in 1 \dots F$, where F is the number of images in the sequence, as:

$$U_p(\alpha) = \sum_{i=1}^{F} \mathcal{R}(\mathbf{w}_{ip}, \mathbf{q}_i^\alpha, \mathbf{t}_i^\alpha, \mathtt{L}_i^\alpha, \mathtt{Q}_i^\alpha, \mathtt{C}_i^\alpha, \mathbf{s}_p) \tag{5.11}$$

which now describes the re-projection error for point p and the model parameters of label α. In practice, we use the adjusted cost used in equation (5.7) which accounts for outliers and assigns different weights to interior points and points in the overlap.

Finally, the fitting of a model to all its assigned points is performed via bundle adjustment optimising the cost defined in equation (5.7) for all the points in the model. As done in Chapter 3 and Chapter 4 we also include temporal smoothness priors on the model parameters.

Once all local regions are reconstructed they must be registered together into the global 3D reconstruction of the non-rigid object (see Section 4.4.1). We note that in this approach we do not perform the refinement step presented in Section 4.4.2, as it is very time consuming in comparison and provides little to no benefit in terms of error minimization.

5.3.3 Neighbourhood Structure

Based on the assumption that our graph should be approximately grid structured, we used the following heuristic: We define an edge E as two points p_1, p_2, and the average 2D distance d,over the whole sequence, between point tracks. We first sort the edges based on this distance, then we traverse this sorted list from smallest to largest, symmetrically adding p_1 to $\mathcal{N}_{p_2}$ and p_2 to $\mathcal{N}_{p_1}$, providing they do not: *(i)* increase the size of a neighbourhood to more than 4; *(ii)* create a triangle, or cycle of 3 nodes in the neighbourhood structure; *(iii)* are not overly large *i.e*. an edge should not span the graph. Providing the overly large edges are discarded before sorting, the procedure is relatively fast and takes approximately half a second to form a neighbourhood structure of 900 points.

5.4 Experiments

We will evaluate our new algorithm on the **Flag**, **Paper** and **Back** data-sets already presented in Chapter 3.

5.4.1 Flag sequence

We begin by providing a quantitative analysis on the motion capture **Flag** sequence, which was already used in Chapter 4 (see Figure 4.1). In Figure 5.5 we show *heat maps* of the log 3D error for each point, where the colours range from dark blue (lowest error) to dark red (highest error), comparing the reconstructions obtained with the Piecewise-Quad algorithm presented in Chapter 4, with the *triangle soup* method [90], and with our formulation described in this Chapter, which we will name NOM-Picewise-Quad. In Table 5.1 we show the numerical results of the relative 3D error, with 3D errors computed as presented in Section 3.4. Note that even though we are using the Quadratic Deformations (QD) model for our local model, as we had done in the Piecewise-Quad

method, once we apply our principled formulation for the choice and optimisation of the patches (NOM-Piecewise-Quad) the 3D reconstruction error drops by a factor of 2. Our new approach also improves [90] by a factor of 1.6 which proves the effectiveness of our approach.

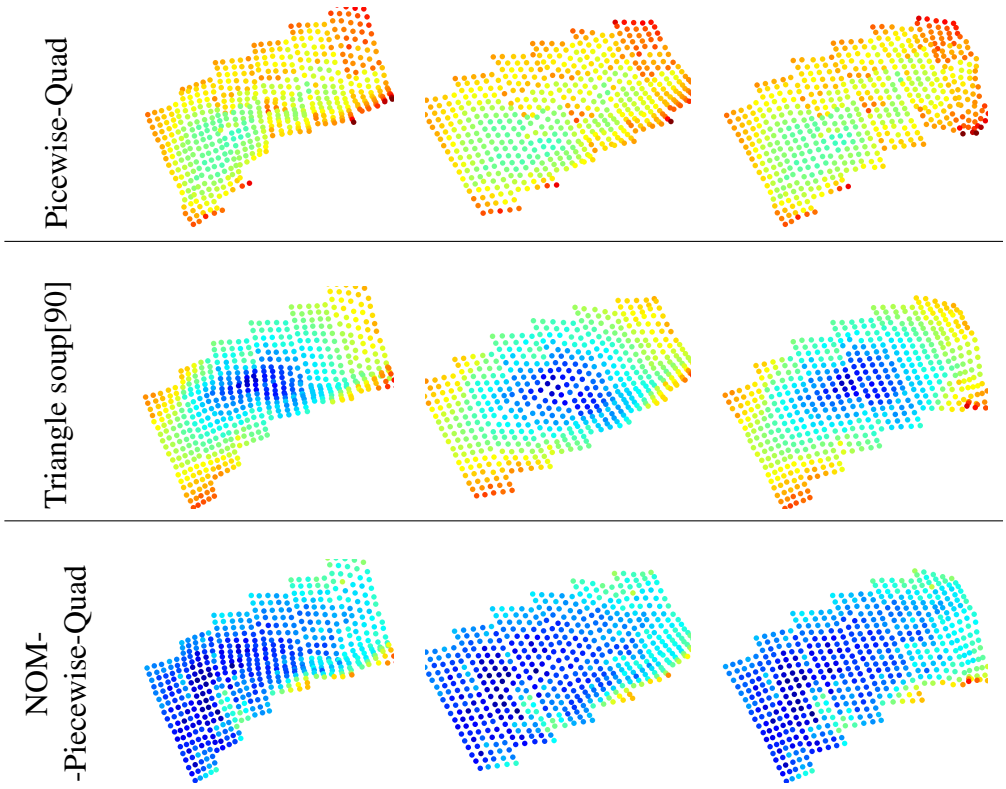


Figure 5.5: A heat map of the log 3D error, on frames 29, 236, and 441 of the flag sequence. The errors vary from dark blue (lowest) to dark red (highest).

5.4.2 Back sequence

We remind that for this sequence we use the stereo reconstruction of [85] as the ground truth 3D values for the points tracked. Figure 5.6 compares the reconstructions from NOM-Piecewise-Quad, *triangle soup* [90] and our Piecewise-Quad method from Chapter 4, by showing the log 3D error as *h*eat maps.

Interestingly, when we evaluated this new sequence with the *triangle soup* method of [90], due to the locally non-rigid motion, most of the points and triangles were

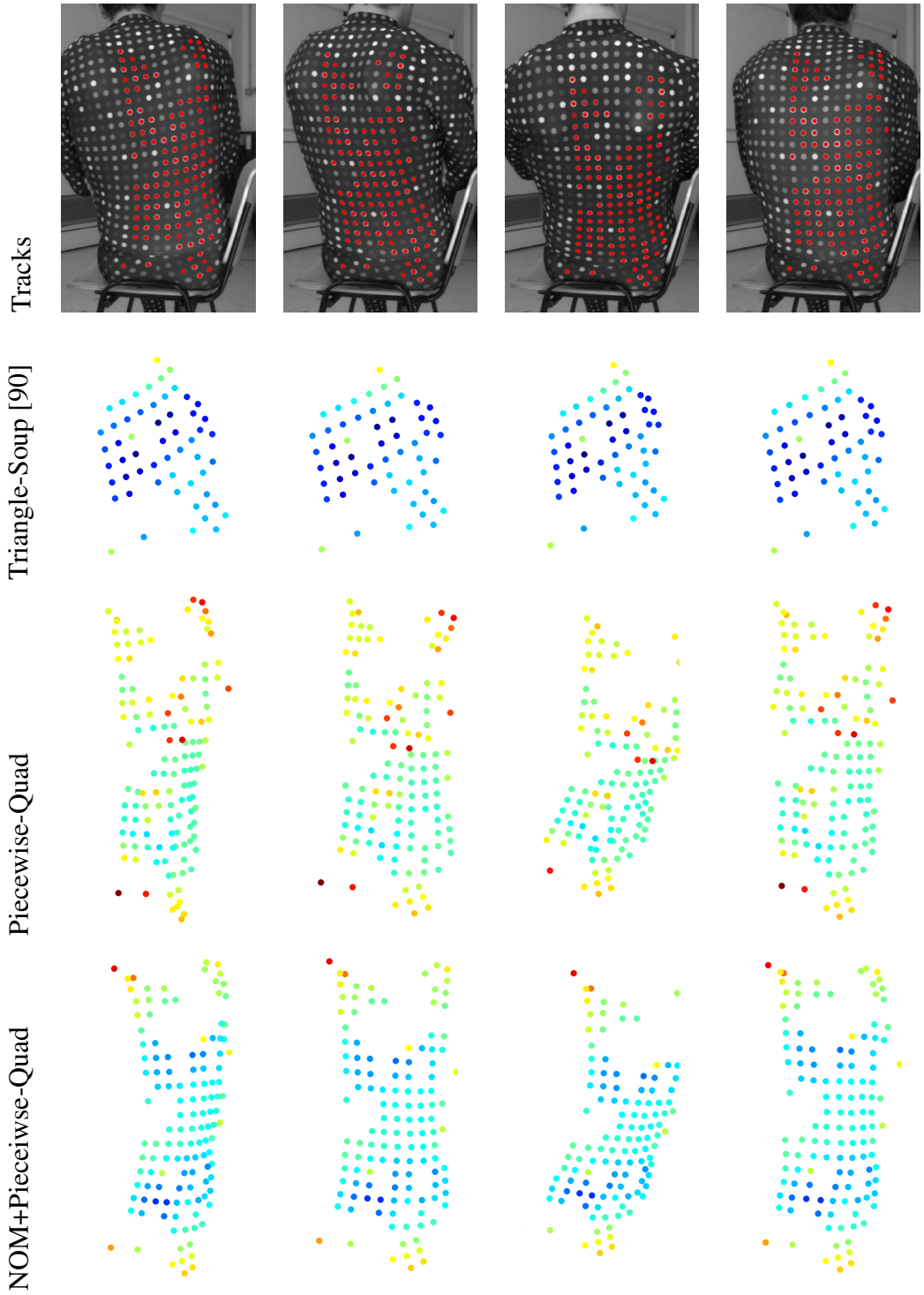


Figure 5.6: A heat map showing the reconstruction and the average log 3D error, on frames 21, 91, 119 and 140 of the Back sequence. The errors vary from dark blue (lowest) to dark red (highest). The second column shows the detail of the lower back, the only area that could be recovered by [90].

Table 5.1: 3D Errors (%) on the **Flag** and **Back** MoCap sequences.

Data set	[36]	[34]	[90]	Our work
Flag	17.09	3.25	2.63	1.59
Back	-	15.20	-	9.17

discarded by the algorithm and the result was a very sparse reconstruction which we were unable to evaluate numerically. Other numerical results for the 3D error can be seen in Table 5.1.

5.4.3 Paper sequence

As was done in Chapter 4, we present a qualitative comparison in the **Paper** sequence between our new NOM-Piecewise-Quad method, the piecewise planar method from Varol *et al*. [100], and the triangle soup form Taylor *et al*. [90]. As also happened on the **Back** sequence, emph triangle soup discarded some triangles as non-rigid, being otherwise a comparable reconstruction. As seen in Section 4.5.3, the piecewise planar algorithm [100] suffers from a lack of smoothness on the surface due to the choice of model.

5.4.4 Choice of models and parameters

As noted previously, our algorithm can be integrated with many different choices of model, and supports the fitting of multiple types of models in the same optimisation. We integrate rigid, and QD model, fitting them as described in section 5.3.2. This is done by alternating between assigning points and fitting models as described in Algorithm 2, but with one important provision. Rather than just refitting one model to each set of points, we fit two models, one linear and one quadratic. We then use the optimisation strategy of Section 5.3.1, to pick a good assignment of models. To compensate for the fact that the QD model always fit regions better than linear models we impose a different MDL cost on each type of model. We use the weighting associated

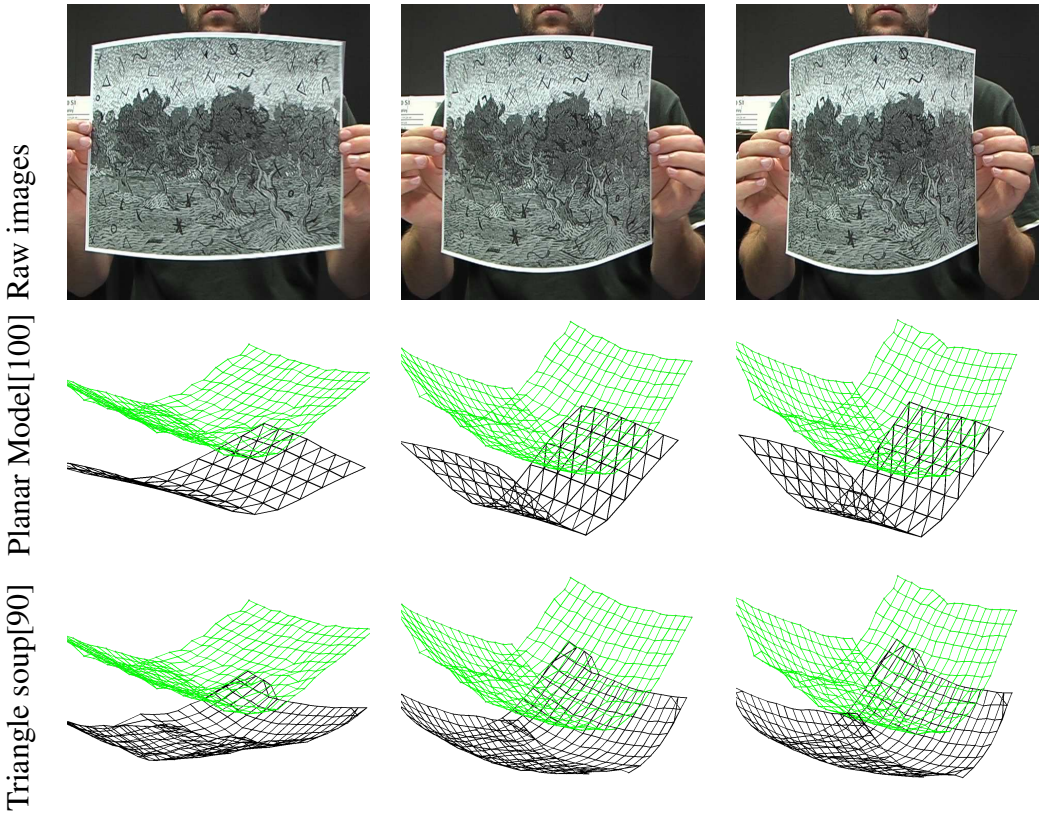


Figure 5.7: Visualization of paper sequence. The 20^{th}, 40^{th} and 60^{th} frames of the paper sequence. The graphics in the two bottom rows show the reconstruction of [100] and [90] (black) overlaid with our results (green). The mesh overlaying our work and of [90] is the neighbourhood structure of section 5.3.3, shown to clarify th structure.

with the Bayesian Information Criterion [29], which suggests that as the intrinsic dimensionality of a quadratic model is three times that of linear model, its MDL penalty should also be 3 times as much. As the ideal choice of absolute MDL cost, and $\lim$ (the truncation value) varies with the amount of noise in the data-set, these are set on a per sequence basis. The initial set of proposed models was formed by fitting a linear model and a QD model to each point and its 9 nearest neighbours and then running the model assignment procedure.

Execution of the graph-cuts stage of our algorithm took approximately 1 second; the fitting of models which was performed with unoptimised Matlab code took approximately 10 minutes. As the code typically took about 3 iterations to converge, the

average run time for fitting around 900 points over 500 frames (the flag sequence) was about 30 minutes. By way of comparison, triangle soup [90] took approximately six hours on the same sequence.

5.5 Conclusion

In this Chapter we showed how the NOM framework from Russell *et al*. [77] can be used as a principled method for adaptive division of a non-rigid object into overlapping patches. These patches can then be reconstructed by our piecewise algorithm presented in Chapter 4 (Piecewise-Quad). The NOM framework formulates the patch division problem as a labelling problem, with the additional property that it enforces patches to overlap, as it is required by our reconstruction algorithm. In our combined framework (NOM-Piecewise-Quad), each label is the set of parameters of a local quadratic model (the choice for the QD model was justified in Chapter 4), and the cost is the re-projection error of the 3D reconstruction with such parameters. By minimising the same cost both in the labelling/patch division step and the patch reconstruction step, we formulate this problem as an alternation optimisation where reconstruction and patch division are performed in turns, which is guaranteed to converge to a local optimum.

We provided experiments on the motion capture data-sets that were used in Chapter 4. Our experiments showed how the NOM-Piecewise-Quad principled formulation for patch division enhances the 3D reconstruction results achieved when using only the Piecewise-Quad algorithm and a manual regular patch division. Additionally, we compared with with other piecewise approaches from Varol *et al*. [100] and Taylor [90], showing that our method provides better quantitative results in these data-sets. We also presented a qualitative comparison on real image sequences.

A summary of algorithms proposed so far is presented in Table 5.2.

Table 5.2: Summary of presented algorithms.

Algorithm	**Piecewise**	**Model**	**Adaptive**	**Initialization**
Quad (Chapter 3)	No	QD	No	Rigid SfM (from first few frames)
Piecewise-Quad (Chapter 4)	Yes	QD	No	Rigid SfM (+ Isomap if known to be flat)
NOM+Piecewise-Quad (Chapter 5)	Yes	QD (supports multiple types)	Yes	Rigid SfM (+ Isomap if known to be flat)

Chapter 6

Dense Non-Rigid Structure from Motion

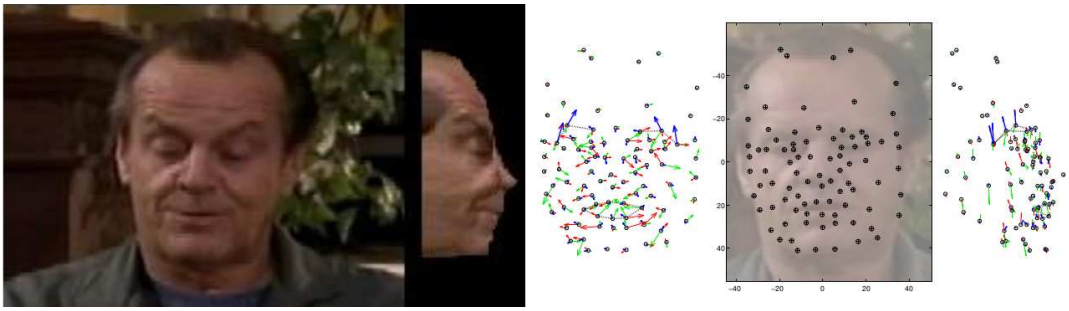
Non-Rigid Structure From Motion (NRSfM) algorithms have reached a degree of maturity that has allowed them to move away from reconstructing simplistic deformations and step up to the challenge of modelling strong, realistic non-rigid motion such as those exhibited by the human body [37] or by a flag waving vigorously in the wind [90, 77].

However, all existing NRSfM approaches are sparse – they scale poorly and can only reconstruct a small number of salient points that are tracked in advance from frame to frame. In this respect, they lie far behind their rigid Structure from Motion (SfM) counterparts which are even capable of a real time dense 3D reconstruction of static scenes that provides accurate depth information for every pixel in the image [67].

Regarding dense 3D reconstruction of non-rigid surfaces from image sequences, Brand's work on 3D morphable models from video [16] is the approach that comes closest to achieving this goal. The algorithm performs simultaneous 3D reconstruction and optic flow estimation by applying the low rank constraint to the 2D correspondences. The strength of this approach is that it does not need 2D tracking data to be provided in advance. Instead, the only inputs to the 3D reconstruction are the image

intensities and their spatial and temporal gradients. The algorithm then computes both the 3D reconstruction and the 2D matching for a sparse set of P points selected in a reference frame. A small regular image patch R is used around the selected points to compute the derivatives and a pure translation model is used for the patch at each point. The focus of this work is on being able to track and 3D-reconstruct non-rigid points with little texture. The results of Brand's approach on a video sequence of 61 frames of an actor talking while moving the head are shown in Figure 6.1.

However, Brand's approach has several drawbacks. First, the optimisation does not include any pairwise smoothness terms. Secondly, although in principle the approach could be applied to all the pixels in the reference frame, this is never demonstrated in practice and instead only a small set of sparse pixels (typically about 100) is reconstructed. Only the sparse points shown in Figure 6.1(b) (90 in this case) are actually reconstructed while the visualization in Figure 6.1(a) is the result of texture mapping the interpolated sparse 3D reconstruction.



(a) (b)

Figure 6.1: (a) One of the images in the input sequence, with the corresponding interpolated 3D reconstruction from [16]. (b) Set of P sparse points used with examples of 3 different deformation modes represented by red, green and blue arrows. Figure from Brand [16].

As discussed in Chapter 2, a further drawback of almost all existing 3D reconstruction algorithms is that they either rely on a known 3D template [21, 83], or need either to estimate a shape basis [18, 96], or rest-shape (Chapters 3 to 5). This reliance on known templates frequently imposes limitations on the kinds of sequences that the

method can be applied to. For example, our approaches discussed over Chapters 4 and 5 perform Isomap upon the first few frames of the video sequence. This relies upon two assumptions: firstly that camera motion in the first frames is substantially greater than object deformation, and secondly that the object being reconstructed can be unwrapped by Isomap *i.e*. that it is a developable surface.

In this chapter we will address these limitations and propose an algorithm for dense, template-free non-rigid reconstruction from video. While this work is the first to perform dense NRSfM, in the sense that every pixel is treated as an individual point, there has been substantial progress in both dense structure from motion (SfM) and sparse NRSfM. Dense approaches to *Multi-view stereo* (MVS) [40, 84], piecewise rigid [41] or live dense reconstruction [67] are able to acquire impressive and accurate 3D models of rigid scenes.

The reconstruction of non-rigid surfaces from monocular sequences remains significantly behind in terms of performance, due to its ill-posed nature – it is equivalent to 3D reconstruction from a single image which cannot be solved without the use of additional priors on the deformations or the camera motion. Our contributions can be summarised as:

A 3D template-free approach: Inspired by Marr's observation that reconstruction from a single camera is essentially a 2.5D problem [65], we recast the problem of NRSfM as the reconstruction of a set of overlapping flexible *surfaces*; We compute a piecewise mapping $f_t(x, y)$ which maps from a location in a reference image $\mathbb{R}^2$ into a 3D location $\mathbb{R}^3$, at time t. This removes the need for a known 3D template or rest-shape and allows the use of one of the images in the sequence as the reference for the $\mathbb{R}^2 \rightarrow \mathbb{R}^3$ mapping.

Dense NRSfM: Building on the work presented in Chapter 5, we adopt a piecewise quadratic approach for 3D reconstruction. The primary bottleneck in the previously presented approaches is the 3D reconstruction of individual quadratic patches. This involves solving a non-linear least squares optimisation problem to minimise the im-

age re-projection error of all the image points belonging to a patch simultaneously. In existing implementations this scales poorly with the number of points. Here, we take advantage of the fact that the predicted location of each point is completely governed by the surface parameters. This observation allows us to integrate over points belonging to the patch, and derive a more efficient cost function that does not use the location of the points directly.

This allows us to replace this computationally intensive optimisation, with a fast, linear time, pre-processing step followed by the minimisation of a quadratic problem of fixed complexity. As a result of these simplifications, the final run-time of our dense NRSfM algorithm is extremely low, and takes approximately 10 minutes to reconstruct a 90 frame sequence of $76,000$ pixels. This compares favourably with existing sparse methods: Our method from Chapter 5 took around 30 minutes to generate a sparse reconstruction (fewer than 0.25% of the points) of the same sequence, while [90] took approximately 7 hours to do the same.

Finally, we provide novel techniques for optimisation: we show how to initialise the Quadratic Deformation (QD) model to avoid poor-quality local minima; and how globally optimal solutions to local sign flip ambiguities can be found efficiently, using pre-existing techniques.

This results in a dense template-free approach that provides complete 3D-models and makes use of all the pixels in the image, bringing NRSfM a step closer to its dense rigid SfM counterparts [40, 84, 41] (see Figure 6.2).

6.1 Problem Formulation

In this chapter we will follow the same inference presented in Chapter 5 where we defined the problem of piecewise 3D reconstruction of deformable surfaces as an alternation between (i) assigning points to local QD models and (ii) fitting of models to the points. This hill-climbing approach is initialised with an excess of models to avoid

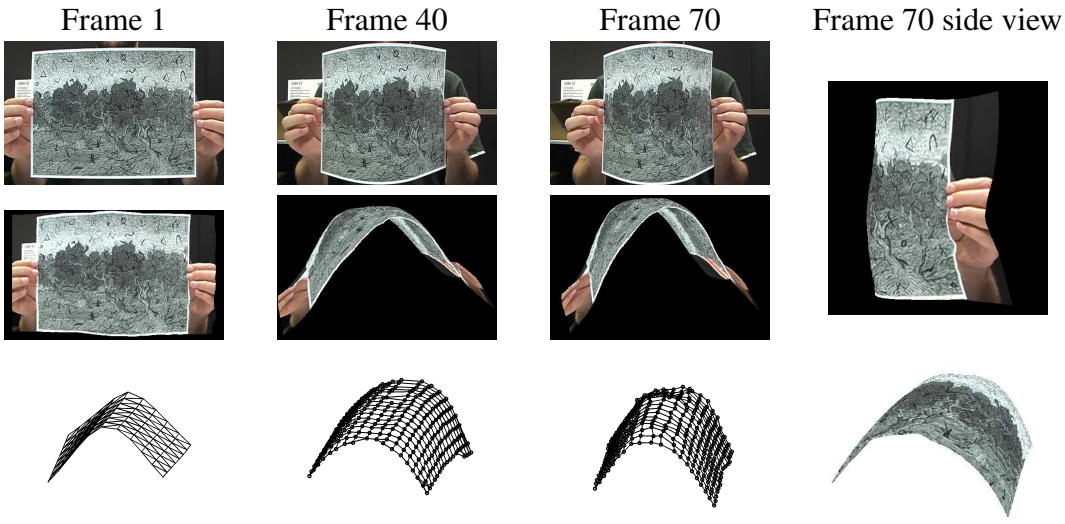


Figure 6.2: **Top:** Input images from the **paper** sequence. **Middle:** Our reconstruction. **Bottom:** Reconstructions with state of the art NRSfM sparse methods [100, 90, 77].

poor local minima.

Our proposed speed-ups to the optimisation of the QD models allows us to use completely dense optical flow as an input resulting in the first approach to NRSfM to estimate completely dense 3D models. Additionally, our approach does not rely on a pre-computed rest shape – instead we directly estimate a mapping from a location in a given image to its 3D location in any frame.

We also modify the assignment stage, by imposing an additional form of pairwise regularisation. This results in an energy to be optimised that contains a unary term expressing the cost of assigning points to models, measured as image re-projection error, a hard constraint enforcing that neighbouring models must overlap, a soft pairwise regulariser, and a minimum description length (MDL) prior [57, 28] used to favour more compact representations that use fewer models.

6.1.1 Global Model Assignment

Consider a sequence of images $\boldsymbol{I}_1, \ldots, \boldsymbol{I}_n$ where $\boldsymbol{I}_r$ is chosen as the reference frame. The input to our algorithm is the dense optical flow field from the reference frame $\boldsymbol{I}_r$

input: Dense optical flow;
Initialise models with an excess of candidate regions;
Fit homographies to models (sec 6.2.1));
Perform overlapping model expansion (sec 6.1.1);
$\Delta = -1$;
while $(\Delta < 0)$ **do**
    CurrentError = GetError();
    Fit QD model to regions (sec 6.2.3);
    Perform overlapping model expansion (sec 6.1.1);
    NewError = GetError();
    Δ = NewError−CurrentError;
end
Flip Patches (sec 6.3.1);
Stitch Patches;

Algorithm 3: Dense NRSfM

to every frame in the sequence. This gives us dense 2D trajectories for every point visible in the reference frame over the entire sequence. We denote this set of points $\mathcal{P}$.

For each image point p we define a 4-connected neighbourhood structure $\mathcal{N}_p$. Given a set of candidate QD models $\mathcal{M}$ (parametrised according to the definition in Equation 6.5) we will estimate the subset of models m_p that each point $p \in \mathcal{P}$ belongs to. We seek the best assignment of a set of models $\mathcal{M}$ to every pixel $p \in \mathcal{P}$ in the image $\mathbf{m} = \{m_1, m_2, \ldots, m_P\}$ such that it: (i) minimises a geometric fitting error and (ii) guarantees that adjacent patches overlap, or more formally, that they share points. Defining the individual cost associated with assigning point p to a fixed model α as $U_p(\alpha)$, Chapter 5 estimated the labelling $\mathbf{m}$ by minimising the following error

$$\underset{\mathbf{m}\in(2^{\mathcal{M}})^{\mathcal{P}}}{\arg\min}\, C(\mathbf{m}) = \sum_{p\in\mathcal{P}} \left(\sum_{\alpha\in m_p} U_p(\alpha) \right) + \mathrm{MDL}(\mathbf{m}), \tag{6.1}$$

We seek a low cost solution that satisfies the constraints

$$\forall p \in \mathcal{P}\ \exists \alpha : p \in I_\alpha, \tag{5.2}$$

and

$$\forall q \in \mathcal{N}_p \wedge q \in I_\alpha \implies \alpha \in m_p. \tag{5.3}$$

Note that this differs from a conventional Markov Random Field in that each point $p \in \mathcal{P}$ is being assigned a set of models $\mathbf{m}_p \in 2^{\mathcal{M}}$, rather than a single model $m \in \mathcal{M}$. The requirement for adjacent models to overlap is expressed in the second constraint if a point p is an *interior* point of a model α (denoted as $p \in I_\alpha$) its neighbours must also belong to that model. Constraint 5.2 enforces that every point must belong to the interior to at least one model.

In practice, our approach from Chapter 5 relied upon neighbouring points being sufficiently far apart as to have substantially different tracks. This created an implicit form of regularisation, that smooths the boundaries of patches. When the tracks are densely sampled from the image, changes between adjacent tracks are much less pronounced and we require additional regularisation to select large regions as belonging to a single model, and to prevent the selection of oddly shaped patches which over-fit to the optical flow.

To do this, we extend cost 6.1 with pairwise potentials defined over the assignment of points to the interior of models. As these pairwise potentials must be defined over *sets of labels* rather than *labels*, they take a non-standard form. If we denote $\mathbf{y}_p$ as the assignment of points to the interior of models, our pairwise potentials can be written as:

$$\psi_{p,q}(\mathbf{y}_p, \mathbf{y}_q) = w_{p,q} \sum_{\substack{\alpha,\beta \in \mathcal{M}, \\ \alpha \neq \beta}} \Delta(\alpha \in \mathbf{y}_p \wedge \beta \in \mathbf{y}_q), \tag{6.2}$$

where $\Delta(\cdot)$ is an indicator function taking value 1 if statement $\cdot$ is true, and 0 otherwise, and $w_{p,q}$ is an image dependent weighting of the pairwise potentials based on the

difference in appearance of the pixels p, and q. This gives a cost of the form:

$$\underset{\mathbf{m}\in(2^{\mathcal{M}})^{\mathcal{P}}}{\arg\min}\, C(\mathbf{m}) = \sum_{p\in\mathcal{P}}\left(\sum_{\alpha\in m_p} U_p(\alpha)\right) + \mathrm{MDL}(\mathbf{m}) + \sum_{\substack{p\in\mathcal{P}\\ q\in\mathcal{N}_p}} \psi_{p,q}(\mathbf{y}_p, \mathbf{y}_q) \tag{6.3}$$

To optimise over this cost function, we note that a minimal cost solution will have each point assigned to the interior of exactly one model. This follows from the proof presented in Chapter 5 that a cost of the form 6.1, will have a minimum in which each point is assigned to the interior of at most one model, and the fact that cost 6.2 is sparsity inducing and will further penalise points belonging to the interior of more than one model. Thus, we will follow the same procedure and minimise 6.1 using a variant of α-expansion [15] defined over interior labels. As we know a priori, each point belongs to the interior exactly one model, the costs of 6.2 can be written in the same form as a generalised Potts model [14], and we augment our previous graph construct with the conventional pairwise potentials used in α-expansion, and solve using graph-cuts [14].

The initial set of candidate models $\mathcal{M}$ is proposed by sampling the image points densely and fitting a model to a small patch around each point. We initialise with an excess of models to avoid convergence to a local minimum, and rely on the MDL prior to remove unnecessary models.

6.2 Template-free Non-rigid Structure from Motion

With the exception of [90] and [100], existing works on NRSfM formulate reconstruction as finding either a sequence of consistent interpolations between static basis shapes [18, 96, 69], or a sequence of deformations of a template [21] or rest shape (Chapter 3 to 5).

As discussed in Chapter 3, the QD model assumes we have prior knowledge of a static rest shape which can be matched under local quadratic deformations to its current shape. To satisfy these assumptions, our approaches from Chapter 3 to 5 required that the deformable shape remains static for the first few frames of the film, while the camera moves. The estimation of the rest shape lies outside the shape fitting optimisation, and if it fails, 3D reconstruction is not possible.

The insight which allows us to eliminate the rest shape is the idea that, in many ways, the QD model is over-expressive. Not only does it encode the location of an observed point in 3D, but it also allows you predict the trajectory of unobserved points, lying in the interior of an object in 3D. For reconstruction from a single view point, this is unneeded. The only question we are interested in asking is:

Given a point p in the reference image, what is its 3D location at time t?

As first observed by Marr [65], this question is inherently a 2.5D one, and best answered by a set of functions $f_t(x, y)$ that map from the image plane $\mathbb{R}^2$ into a 3D location $\mathbb{R}^3$, at time t. If the object we are modelling has hard edges, f_t is unlikely to be smooth, while if we are modelling multiple disjoint objects, f_t need not be continuous. However, given dense real world data, f_t will be piecewise smooth, and can be approximated by decomposing the image plane into a set of regions, and using a local quadratic function to approximate f_t for each region.

We will keep making use of the QD model as people are highly sensitive to sudden changes in the gradient of reconstructed surfaces, and to avoid these sudden changes, we must use piecewise models who's gradient can vary. QD models are the simplest polynomial with a variable gradient, and their use provides a balance between robustness via not over-fitting, and the avoidance of visible artefacts in the reconstruction.

The problem of simultaneously estimating local QD model and regions is challenging. However,we have already shown in Chapter 5 that the combination of graph-cuts, and greedy model fitting are well suited for such problems.

6.2.1 Quadratic Local Model Fitting

As seen in Chapter 4, shape fitting for each individual patch is formulated as a problem of non-linear least squares regression. The objective, which we seek to minimise, over all points belonging to the model, takes the per-point form:

$$U_p(\alpha) = \sum_{i=1}^{F} \mathcal{R}(\mathbf{w}_{ip}, \mathbf{q}_i^\alpha, \mathbf{t}_i^\alpha, \mathtt{L}_i^\alpha, \mathtt{Q}_i^\alpha, \mathtt{C}_i^\alpha, \mathbf{s}_p), \tag{5.11}$$

where $\mathcal{R}$ is the re-projection error:

$$\mathcal{R}(\mathbf{w}_{ij}, \mathbf{q}_i, \mathbf{t}_i, \mathtt{L}_i, \mathtt{Q}_i, \mathtt{C}_i, \mathbf{s}_j) = ||\mathbf{w}_{ij} - \hat{\mathbf{w}}_{ij}||^2 = ||\mathbf{w}_{ij} - \Pi\mathtt{R}_i(\mathbf{q}_i)\,[\mathtt{L}_i\,\mathtt{Q}_i\,\mathtt{C}_i]\,\mathbf{s}_j - \mathbf{t}_i||^2 . \tag{3.12}$$

Since in this chapter we seek a mapping $f_t(x, y)$ from location in a reference image $\mathbb{R}^2$ into a 3D location $\mathbb{R}^3$, we will implicitly assume $\mathbf{s}_p = [x, y, x^2, y^2, xy]^T \in \mathbb{R}^5$ as there is no need to define the terms corresponding to the z coordinate of the shape. $\mathtt{L}_i$ and $\mathtt{Q}_i$ will now be 3×2 matrices, while $\mathtt{C}_i$ is a 3-vector. However, for the sake of simplicity we will keep the same notation as this problem is equivalent to fixing the z coordinate in the formulation of previous chapters to a constant value. We note that x and y are now the image coordinates of point p in the reference image.

We choose these components of $\mathtt{A} = [\mathtt{LQC}]$ by initially approximating the 2D tracks belonging to each patch as being the orthographic projection of a local rigid plane. Then we define these components of $\mathtt{A}$ as corresponding to the mapping from x and y in the reference image to coordinates of the rigid plane. Unlike template based approaches, we allow this method to fail and occasionally to produce bad estimates. Any bad models proposed will have a high re-projection error, and will be discarded by the graph-cuts optimisation.

6.2.2 Initial Model Estimation

To initialize our deformation coefficients, we start by applying [75] to the tracks contained in local 10 by 10 pixel patches, which returns an embedding of those tracks into 2 dimensions. Lets denote this embedding as $\mathtt{I}'_j = [x'_j; y'_j]^T$ for every point j, whereas the coordinates of point j in the reference image are denoted by $\mathtt{I} = [x_j; y_j]^T$. Since $\mathtt{I}'$ was computed just for a small patch of 10 by 10 pixel, it is not practical to use this matrix to compute our augmented shape matrix $\mathtt{S}$, in line to what was done with Isomap in Chapter 4. As our model fitting approach requires to evaluate each model on every point we want to reconstruct, it is advised to have a common representation for all the points, which is chosen to be their 2D coordinates in the reference image. We thus use the information in $\mathtt{I}'$ as a way to initialize the first two rows of the linear deformation matrix $\mathtt{L}_i$ at each frame, by computing the 2×2 such that $\mathtt{I}' = \mathtt{L}_{1:2,:}\,\mathtt{I}$, where $\mathtt{L}_{1:2,:}$ denotes the sub-matrix of the first two rows of $\mathtt{L}$, and the frame index i was dropped for notation simplicity. The rigid motion of each patch $\{\mathtt{R}_i, \mathbf{t}_i\}$ is then initialised using [68].

Treating $\mathtt{A}$ as fixed, the initial motion and planar shape parameters for each patch are refined using bundle adjustment [99] to minimise the following cost function:

$$\min_{\mathtt{R}_i, \mathbf{t}_i, x, y} \sum_{i=1}^{F} \sum_{j=1}^{P} ||\mathbf{w}_{ij} - \Pi \mathtt{R}_i \mathtt{A}_i [x_j \; y_j]^T - \mathbf{t}_i||^2. \tag{6.4}$$

Both the warping techniques of [75] and bundle adjustment scale poorly with the number of points in the models. However these optimisations are done once for each initial patch proposal, which are usually very small and consequently does not slow the overall optimisation significantly.

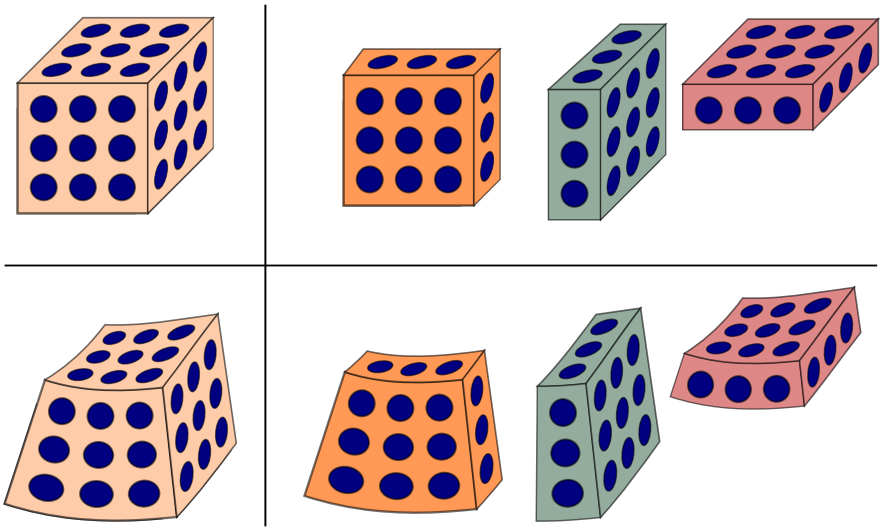


Figure 6.3: The use of local quadratic deformations with a rest shape is more robust to arbitrary choices of patches, while our surface based models require a good assignment of points to models **Leftmost:** A deformation can be represented by a single quadratic deformation of the rest shape, that maps from $\mathbb{R}^3 \rightarrow \mathbb{R}^3$. **Rightmost:** Modelling the deformation as a set of surfaces requires points to be correctly assigned to three separate models. Describing the object as a set of overlapping smooth surfaces becomes an increasing accurate approximation as we increase track density.

6.2.3 Fast Dense Fitting of the Quadratic Model

The QD model of an individual patch α can be parametrised as $\alpha = \{\mathtt{A}^\alpha, \mathtt{R}^\alpha, \mathbf{t}^\alpha\}$. Adopting index $j \in \alpha$ for points in the reference image belonging to model α, we use $\mathbf{w}_{ij}$ to denote the projection of point j in frame i. We define the per-frame/model cost $C^{i,\alpha}$ as

$$C^{i,\alpha}(\mathtt{A}_i, \mathtt{R}_i, \mathbf{t}_i) = \sum_{j \in \alpha} ||\mathbf{w}_{i,j} - \Pi \mathtt{R}_i \mathtt{A}_i \mathbf{s}_j - \mathbf{t}_t||^2, \tag{6.5}$$

being the aggregate cost for all the points belonging to model α in frame t. Evaluating the cost in this form requires computing a cost for every combination of point j and frame i. In the dense case the number of points to reconstruct can be several orders of magnitude higher than in the sparse cases studied on previous chapters, rendering such algorithms impractical. However, our formulation ensures that the matrices S for every patch are constructed from the reference image and are thus a constant factor in our

optimisation. Making use of this observation, it is possible to redefine Equation 6.5 as

$$
\begin{aligned}
C^{i,\alpha}(\mathtt{A}_i, \mathtt{R}_i, \mathbf{t}_i) = & \sum_{j \in m} ||\mathbf{w}_j||^2 + tr(\mathtt{RA}(\sum_{j \in m} \mathbf{s}_j \mathbf{s}_j^\top)(\mathtt{RA})^\top) + \sum_{j \in m} ||\mathbf{t}||^2 \\
& - 2\langle \sum_{j \in m} \mathbf{w}_j, \mathbf{t} \rangle - 2tr((\sum_{j \in m} \mathbf{w}_j \mathbf{s}_j^\top)(\mathtt{RA})^\top) \\
& + 2\langle \sum_{j \in m} \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t} \rangle,
\end{aligned}
\tag{6.6}
$$

where, the summation over j can be separated from non-rigid motion parameters in $\mathtt{R}$, $\mathtt{A}$ and $\mathbf{t}$, revealing such constant factors (for details on the derivation, see Appendix A). This new formulation allows us to pre-compute the summations over j before optimising the model parameters, which in turn makes our optimisation step independent of the number of points to reconstruct. In the case where $\mathcal{P} \gg \mathcal{F}$, the usual scenario for the dense 3D reconstruction problem, the efficiency gained in the optimisation step overcomes the added cost from performing the precomputation of the terms depending on j. It is this observation that provides the key to performing dense NRSfM.

6.3 Post Processing

As each patch is reconstructed in its own reference system we must resolve ambiguities inherent to orthographic cameras: translation in the Z axis and reflection ambiguities.

6.3.1 Flip Resolution

As discussed in Section 4.4.1, the 3D reconstruction from an orthographic camera carries ambiguities regarding the relative translation along the Z axis, and a sign ambiguity on the reconstructed depths, making it impossible to determine if an object is either convex or concave without prior knowledge. Although solving this problem is NP-hard, in Section 4.4.1 a greedy heuristic algorithm was proposed to solve this problem with satisfactory results when considering sparse data. However, we experi-

enced problems when using it with our dense patch reconstructions and thus resort to a different approach to solve this problem. The main difference between the sparse and dense cases is in the real surface area corresponding to the overlapping regions. Considering the object area corresponding to a two point width overlap in the sparse case, to achieve the same overlap area in the dense case far more points would be needed, resulting in a prohibitive increase in complexity in the NOM approach. By relying on smaller overlap area, it becomes harder to disambiguate the correct flips, and so we must use a method that takes more information from neighbouring patches into account.

Taylor [90] proposed solving the NP-hard problem of flip resolution using a combination of fusion moves [59], and heuristic move proposals. Following Taylor [90] we consider a flip cost $F(\mathbf{z})$:

$$\underset{\mathbf{z}\in\{-1,+1\}^{\mathcal{M}\times\mathcal{F}}}{\arg\min} F(\mathbf{z}) =$$

$$\sum_{i\in\mathcal{F}}\sum_{\alpha,\beta\in\mathcal{M}}\sum_{j\in\alpha\cap\beta}||\nabla^i_\alpha(j)z_{\alpha,i}-\nabla^i_\beta(j)z_{\beta,i}||_2^2 \quad +\lambda\sum_{\alpha\in\mathcal{M}}\sum_{\substack{i\in\mathcal{F},\\ i\geq 1}}||\nabla^i_\alpha(j)z_{\alpha,i}-\nabla^i_\alpha(j)z_{\alpha,i-1}||_2^2$$

where $\mathcal{M}$ is the set of models, $\mathcal{F}$ the set of frames, and $\mathbf{z}$ describes the set of proposed flips or sign changes, $j\in\alpha\cap\beta$ is a point j lying in the overlap of the points belonging to models α and β, and $\nabla_\alpha(j)$ is the gradient of the depth of point j according to the quadratic model α.

While, as Taylor [90] noted, this formulation of resolving patch flips is NP-hard, in practice we observed that a globally optimal solution was always found by using a combination of QPBO [11] with the probe technique [12], as implemented by [56].

Owing to the relatively small number of patches[1], a consequence of active assigning of points to patches, using the techniques of [77], and a better choice of optimisation technique, flip resolution took approximately one minute to converge.

[1] In a typical problem, see Section 6.4, a reconstruction uses fewer than 70 active models.

6.3.2 Global Shape Ambiguities

The globally optimal solution found in the previous section can still suffer from ambiguities. In [90], the authors observed that it was not possible to resolve the ambiguity between an 'S'-shaped developable surface, and a concave or convex surface, and suggested user intervention to resolve this ambiguity (see Figure 6.4). These ambiguities are a limitation of the orthographic camera model, and thus require prior knowledge to be resolved. Instead of relying a direct user intervention, the desired shape by applying a global sign flip to the depth of the points in the object as a post processing step.

However, in the case the object deforms by going from convex to concave (or vice-versa) our prior will guide the deformation back to the convex state after reaching the "middle" point. In this case, the user would be require to define which section of the sequence the object remains convex, and which section the object will be turned concave via means of the depth sign flip.

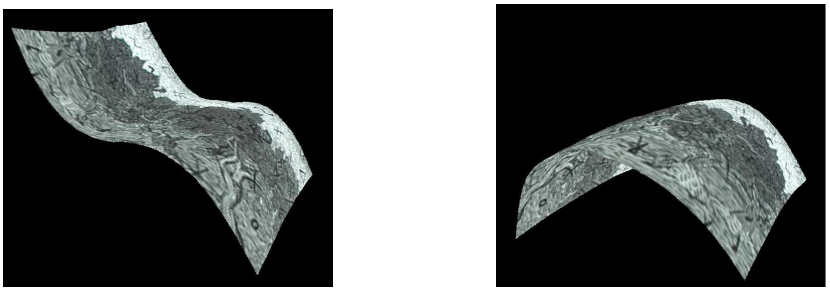


Figure 6.4: **Left:** The monotonic increasing solution found using graph-cuts. **Right:** The solution found with convex priors.

In this section, we will focus on the gradient with respect to x, ∇_x. Naturally, the same potentials would also be generated for the gradient with respect to y, and summed.

Two possible approaches suggest themselves for encouraging solutions found to be convex. We could modify the pairwise costs to be of the form

$$||\nabla^i_{x,\alpha}(j)z_\alpha - \nabla^i_{x,\beta}(j)z_\beta - k||^2_2, \tag{6.7}$$

where k expresses a preference that the gradient of model α be k smaller than the gradient of model β. This matches the definition of convexity, as a twice-differentiable function is convex *if and only if* its second derivative is non-negative, and an appropriate $k_{i,j}$ encourages the change of gradient between overlapping patches to be negative.

Alternatively, we may use a unary potential to express a weak expectancy that the gradient with respect to x of those patches on the left side of the image to be increasing and decreasing on the right hand side of the image. These potentials, based on the gradient of a Gaussian, take the form:

$$U(z_\alpha^i) = -\gamma(j-\mu)\exp(-\sigma^2(j-\mu)^2)\nabla_\alpha^i z_\alpha^i \tag{6.8}$$

where γ, and σ are arbitrary constants governing the strength and range of the prior. Of the two approaches, the pairwise convex prior was found to overly smooth most sequences, if it was strong enough to enforce convexity. Instead the second unary based prior was uniformly imposed on all sequences.

Resolution of the translation/depth ambiguity We follow our approach described in Chapter 4 and use the shared points in the overlapping region to align the patches along the Z axis since their 3D coordinates should agree. We perform a per frame greedy stitching where the depth of single patches are iteratively fixed to minimise the sum of squared distances between the depths predicted by the current patch and the predictions of the fixed patches.

Interpolation Even after performing flip resolution and depth alignment, local models still disagree about the precise location of points in the overlap, and making a hard assignment of point to models leads to discontinuities in the surface. To eliminate discontinuities in the reconstruction, we estimate the depth of each point as a weighted average between the depth estimate of each model. These per model weights are cho-

sen as the inverse of the L_1 distance of the point we are averaging from the nearest point belonging to the interior of the model, with the distances being computed per patch, using the fast distance transform of [38].

6.4 Experimental evaluation

Our approach to dense NRSfM requires dense pre-computed tracks. We make use of multi-frame optic flow algorithms [42, 92] to extract these from video. One of the major difficulties we faced was in how to evaluate the quality of dense NRSfM, as there are very few videos of non-rigid moving objects with dense ground truth available. As such, the majority of our evaluations are qualitative rather than quantitative.

Figure 6.2 shows 3D reconstructions of the **Paper** sequence of [100] and a comparison with existing sparse reconstructions. In Figure 6.6 a reconstruction of a face sequence from the TV series *LOST* is shown.

We evaluate our algorithm on a synthetic variant of the 540-point 3D **Flag** sequence [34]. In [42], the authors synthetically interpolated this sequence with b-splines to create a denser 9,620 point sequence. This sequence is projected into a top-down view, and we reconstruct this dense sequence in 3D. Renderings of the ground-truth and our reconstruction from a novel-view point can be seen in Figure 6.5. We obtain 4.72% error on this dense sequence, vs. the reported errors of 3.25% of our approach from Chapter 4, 2.63% of [90], and 1.59% of our approach from Chapter 5 on the sparse sequence.

Even though our dense NRSfM algorithm takes advantage of more data points (*i.e.* more information) its 3D reconstruction error is 2 to 3 times higher than the sparse state of the art approaches. As we move from sparse to dense reconstructions, such increase in 3D reconstruction error can originate from a failure in one or several of our algorithm components. Possible causes for failure could be:

1. Poor quality 3D reconstruction of each patch by the model fitting step.

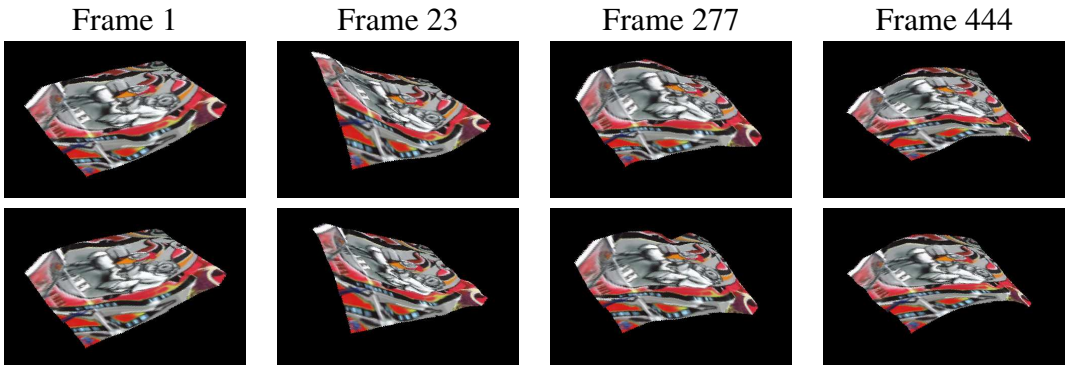


Figure 6.5: Reconstruction of the dense 9,000 point **flag** sequence based upon a top down orthographic projection. **Top:** Ground truth motion of the **flag**. **Bottom:** Our reconstruction.

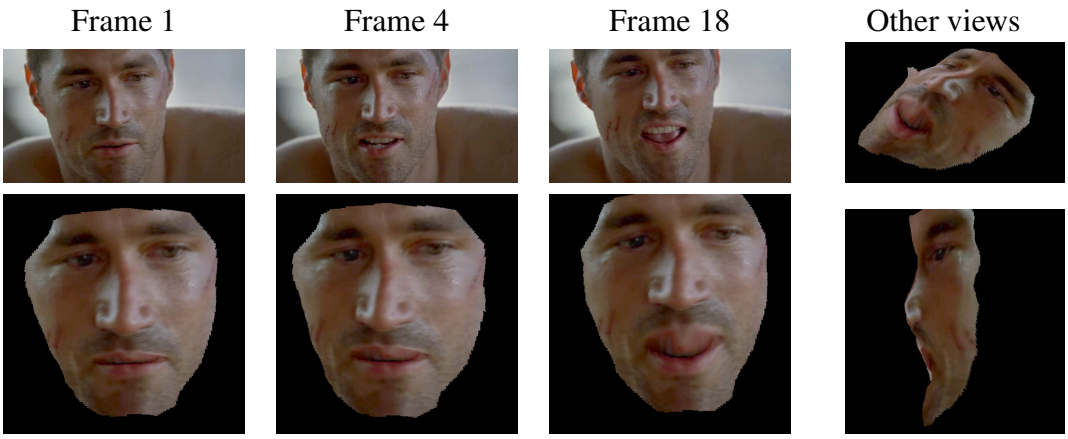


Figure 6.6: **LOST** sequence. Despite restricting the algorithm to a sub-sequence containing only minor rotations (this is required to preserve optic flow), we are able to reconstruction the face including the nose.

2. Overfitting of the QD model due to complex object boundaries.

3. Difficulties in correctly resolving the convex/concave and depth ambiguities during patch registration.

To better understand the cause of this discrepancy, our experiments aim at isolating the contributions of these three factors. We perform a side by side comparison of the dense NRSfM algorithm presented in this chapter and the sparse NRSfM algorithm presented in Chapter 4. To simplify, we begin our analysis by reverting to the regular patch division proposed in Chapter 4.

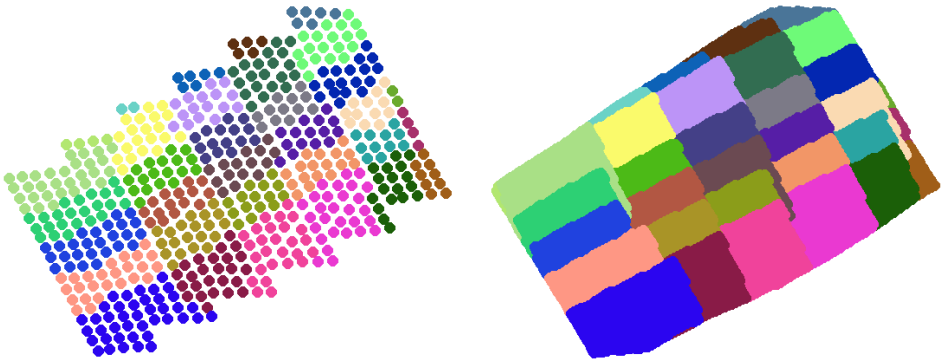


Figure 6.7: **Left:** Illustration of the division into 36 patches of the **Flag** sequence used in Chapter 4, where each colour represents a different patch. **Right:** Illustration of the **dense Flag** sequence with the patch division corresponding to the sparse division of Chapter 4.

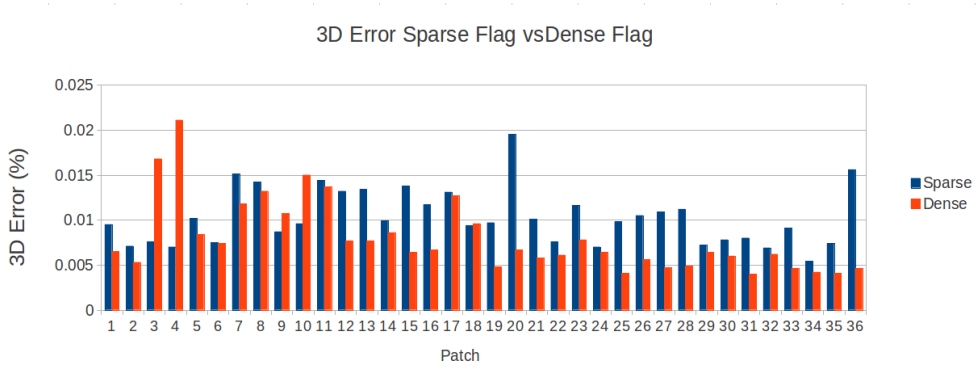


Figure 6.8: Comparison of the 3D reconstruction error per patch, normalized by the Frobenius norm of the full data matrix.

Patch reconstruction Given the regular division of Figure 6.7, each dense patch is initialized and reconstructed with the dense formulation presented in this chapter. It is not expected that every patch should have the same 3D reconstruction error for the sparse and dense cases. Such measures should, though, be comparable. As expected, the use of dense information generally provides a small boost in the quality of patch reconstruction, although this need not always be the case (see Figure 6.8 for a per patch comparison of the 3D reconstruction error and Table 6.1 for per point mean error and standard deviation, normalized by the Frobenius norm of the full data matrix).

Table 6.1: Summary of the 3D reconstruction errors per patch for the sparse and dense flags.

Sequence	**Average Error (%)**	**Standard Deviation (%)**	**Maximum Error (%)**	**Minimum Error (%)**
Sparse Flag	0.0103	0.0031	0.0195	0.0054
Dense Flag	0.0080	0.0040	0.0221	0.0040

Patch division using NOM formulation To test if our formulation for adaptive division leads to overfitting, we reconstruct the patches as described in this chapter, but use the ground truth data information to resolve a single concave/convex ambiguity, and the patch depth ambiguity. Table 6.2 presents a summary of the comparison of the alignment methods for five different reconstruction scenarios for the dense **Flag** sequence: simple reconstruction of the regular patch division from Chapter 4; a single iteration of our dense NRSfM algorithm presented in this chapter after initializing with the regular patches from Chapter 4; multiple iterations of our dense NRSfM method after initializing with the regular patches from Chapter 4; and the complete pipeline presented in this chapter for two different values of the MDL cost – 10^6 and 10^5.

As can be seen in Table 6.2, when correctly aligning the dense patches the reconstruction is considerably lower. Additionally, the lowest 3D reconstruction error reported in Table 6.2 (1.18%) is comparable to our state-of-the-art sparse reconstruction results presented in Chapter 5 (1.59%). We conclude that our formulation provides very accurate 3D reconstructions of the local regions. However, the alignment of the patches into the final reconstruction is deficient, as it performs poorly when compared to the sparse case.

Patch alignment and stitching Comparing the patches obtained by regular division (from Chapter 4) and the patches obtained by the adaptive division with the NOM

Table 6.2: 3D reconstruction error of our registration algorithm vs. alignment to the ground truth data.

Reconstruction Conditions	**Our Alignment 3D error (%)**	**Ground Truth Alignment 3D error**
Regular Patches (No patch optimisation)	3.55	2.13
Regular Patches (single NOM iteration)	4.12	2.12
Regular Patches (NOM patch optimisations)	4.14	1.71
Dense NRSfM ($\mathrm{MDL} = 10^6$)	8.42	2.93
Dense NRSfM ($\mathrm{MDL} = 10^5$)	4.72	1.18

formulation, the later have a much smaller area of overlap even when increasing the size of the local neighbourhoods. Consequently, we must analyse how well our alignment methods scale to dense data, and how the area of overlap influences the final 3D reconstruction results.

When aligning the reconstruction of dense patches given by regular division (with large overlap area), our 3D reconstruction error for the dense case (3.55%) is comparable to the value obtained for the sparse algorithm in Chapter 4 (3.25%). This shows that our method has no problem scaling to dense patches, provided the overlapping area remains the same.

In the second row of Table 6.2 we show the effect of transforming the patches obtained from the regular division into a division returned by the NOM formulation. The effect of the reduction in the overlapping area is an immediate increase in the 3D reconstruction error. This is caused by the difficulty in correctly aligning the patches, given such a small area of overlap. Still, as the alignment to the ground truth data shows, our local reconstructions retain the same quality, with the 3D reconstruction error practically unchanged. These results, together with the 3D reconstruction errors found in

other scenarios shown in Table 6.2, show that our alignment algorithm breaks down in performance if the area of overlap greatly decreases, as the available information for resolving the ambiguities is too small.

Conclusions of sparse vs dense algorithm comparison After analysing our three possible causes for the breakdown in performance of our dense NRSfM algorithm presented in this chapter, we can safely conclude that, individually, every step of our previous sparse formulation can scale to the dense case. We obtained very accurate local 3D reconstructions, with errors comparable to the state-of-the-art sparse cases when ground truth alignments are provided. However, the area of overlap returned by the NOM formulation proved to be too small for our proposed algorithm to resolve the patch ambiguities and correctly stitch them together. Our experiments show that correct alignment is possible if the area of overlap is increased. A possible solution to this breakdown in performance, without changing the NOM formulation, is to perform the multiple model assignment based on *superpixels*, instead of each pixel in the image. This would simultaneously reduce the number of required variables, decrease the run-time of the optimisation and increase the overlapping area of neighbouring model assignments.

6.5 Conclusion

In this chapter we present an approach to perform dense non-rigid structure from motion, and we show how the QD model can be used for template-free reconstruction. In breaking this new ground, we found several technological hurdles that had to be overcome.

We modified the formulation presented in Chapter 5 to improve the regularisation of patches formed from dense tracks; and proposed a novel pre-processing step to allow the fast fitting of quadratic models; we showed how local minima in QD model

may be avoided by better initialisation; and we showed how the problem of patch resolution, previously been solved using heuristics can be solved exactly using existing techniques.

Our results show a substantial qualitative boost over existing sparse reconstructions, and gives vivid reconstructions on real world sequences. However, our reported error on the **dense Flag** sequence is 3 times higher than the state of the art sparse methods. After comparing the dense and sparse algorithm step by step, we showed that locally our reconstructions are still very accurate. However the breakdown in performance arises from poorer patch registration and stitching, which is caused by a relative smaller are of overlap between the patches when compared to the sparse case. A solution to this problem can be to perform our formulation on superpixels, effectively decreasing the number of point to label and increasing the area of overlap.

A summary of the algorithms proposed throughout the chapters is presented in Table 6.3.

Table 6.3: Summary of presented algorithms.

Algorithm	Piecewise	Dense	Model	Adaptive	Initialization	Missing Data
Quad (Chapter 3)	No	No	QD	No	Rigid SfM (from first few frames)	Can lose tracks in $\mathtt{S}_q$ Cannot incorporate new tracks
Piecewise-Quad (Chapter 4)	Yes	No	QD	No	Rigid SfM (+ Isomap if known to be flat)	Can lose tracks in $\mathtt{S}_q$ (per patch) Cannot incorporate new tracks
NOM+Piecewise-Quad (Chapter 5)	Yes	No	QD (supports multiple types)	Yes	Rigid SfM (+ Isomap if known to be flat)	Can lose tracks in $\mathtt{S}_q$ (per patch) Cannot incorporate new tracks
NOM+Piecewise-Rigid (Chapter 7)	Yes	No	Rigid	Yes	Rigid SfM	Can lose and incorporate new tracks
Dense-Piecewise-Quad (Chapter 6)	Yes	Yes	QD	Yes	Per patch Unwrap Mosaic + BA on affine motion	Can lose tracks in $\mathtt{S}_q$ (per patch) Cannot incorporate new tracks

Chapter 7

Networks of Overlapping Models for Articulated Structure from Motion

One problem of particular interest in the computer vision community is human motion analysis. Estimating the 3D pose of the human body purely from image data has important applications ranging from bio-mechanics to cinema post-production, computer gaming, animation and human behavior analysis. Following the theme of this thesis, we focus on the specific case of full 3D reconstruction using only the 2D positions of P interest points tracked over time, along a sequence of F images acquired under orthographic viewing conditions.

Most algorithms for 3D pose estimation of articulated bodies require prior knowledge of a model of its underlying structure, usually given by a kinematic chain [86, 19, 89], which requires manual intervention to create. This high level of intervention is undesirable in many circumstances. For example in animation or gaming, an actor should be able to pick up and interact with a rigid object, effectively augmenting their skeletal structure, without the need for a graphical artist to generate a new model. Given this predefined 3D skeleton model, these approaches track the articulated motion, estimating the joint angles, but do not usually recover the full 3D shape of the object. Instead, we take a different approach and demonstrate how the estimation of

the full 3D shape, motion, and underlying skeletal structure of one or more articulated bodies can be derived directly from 2D correspondences in a video sequence acquired with a single camera without the need for any prior models.

Articulated motion is typically formulated as a special case of NRSfM where the non-rigid bodies are seen as a set of rigidly moving links connected by articulations (or joints). Previous approaches to recovering both articulated structure and motion purely from 2D tracking data include factorization methods [106, 98] which model articulated motion as a set of intersecting motion subspaces. These methods require two steps: first a motion segmentation algorithm separates the 2D trajectories into different articulated parts; and second a factorization approach is used to estimate joint positions and articulation axes. Yan and Pollefeys [106] follow this with a third step that builds the kinematic chain automatically from the segmented subspaces. Each articulated part can be recovered as a rigid shape using the factorization method [93].

Such pipeline approaches are inherently unstable. A failure in any of the early stages of reconstruction cannot be recovered from, and such difficulties are often unapparent until the final reconstruction fails. As an alternative to this multi-stage formulation, we propose an algorithm which performs a simultaneous decomposition of the articulated body into its constituent parts and reconstructs the full 3D shape of the object, revealing its skeletal structure.

Following the trend of energy-based multiple model fitting described in Chapter 5, we tackle articulated reconstruction from 2D tracks using a piecewise approach. We keep the assumption that an articulated object can be approximated by a set of rigid segments linked by articulations. We also assume that these articulations can be described as the overlap between rigid links. The key idea is to segment the object into its constituent rigid segments, while enforcing overlap between neighboring segments. Similarly to other piecewise approaches [100, 90, 22, 34, 77], segments are reconstructed independently and points in the overlapping regions are then used to stitch them together to create a full 3D articulated body.

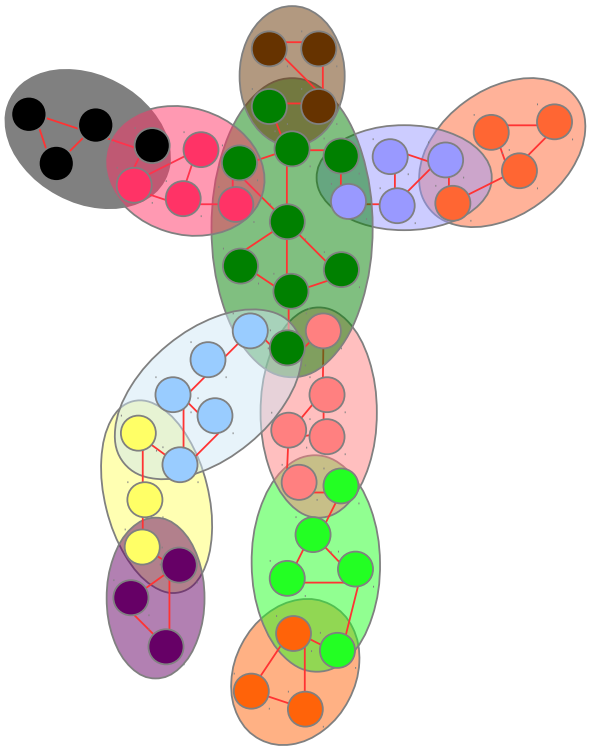


Figure 7.1: Example of the overlapping model assignment. Each circle represents a point. Each colour represents an interior point lable. Red lines represent the neighbourhood connections. Coloured ellipses represent the overlapping model assignment.

As seen in Chapters 4 to 6, piecewise solutions have been applied with success to deformable surface reconstruction [100, 90, 34, 77]. In this chapter we demonstrate that they are equally applicable to the problem of Articulated Structure from Motion (A-SfM). Our approach distinguishes itself from these as, on articulated data, it estimates semantically meaningful rigid parts and gives the location of joints, rather than returning surface regions. Compared to the algorithms described in Chapter 4 and Chapter 5 this approach does not require an initial estimate of a rest shape.

As discussed in Chapter 5, the strength of our approach [77] comes from viewing both the decomposition into parts and the 3D reconstruction as the optimisation of a single cost function, namely the image re-projection error, subject to a spatial constraint that neighbouring points should also belong to the same model. This gives us the ability to switch back and forth from the assignment of points to parts, and fitting

a rigid model to the parts, in a hill-climbing approach, allowing us to recover from previous mistakes and refine our current model estimates as we go.

Two significant advantages of our formulation over previous motion segmentation algorithms [101, 32] are: (i) that it does not require the number of motions to be known in advance; (ii) we exploit the spatial prior that points which are physically close are likely to belong to the same model. Our only necessary assumptions are that we find a minimum of three tracked interest points on each rigid part, which is needed to perform 3D reconstruction, and that at least one point is located in the intersection of body parts — this last constraint is due to the fact that we rely on points belonging to multiple models to guarantee the spatial consistency of the global 3D shape. Both of these constraints are guaranteed by our inference model, provided that each point has at least two neighbours, and that the graph of points in the human skeleton is path connected. See section 7.1.1 for more details.

7.1 Problem Formulation

The typical framework of A-SfM methods stems from the Tomasi and Kanade [93] paradigm: an *articulated object* described by a set of $\mathcal{P}$ point tracks, observed by an orthographic camera in a sequence of F image frames. We assume this articulated object can be accurately approximated by a set of rigid segments that form an articulated forest[1]. We make no assumption about the number of segments of the object nor which feature points belong to each segment. Our goal is to recover the 3D coordinates of the corresponding 2D point tracks, given the assumption of articulated motion.

Given either the model parameters, or the assignment of points to models, the problem of reconstruction is straightforward. Given an assignment of points to rigid models

[1]Again, this is a simplifying assumption, the fact that the graph formed is a forest (*i.e.* contains no cycles) is not used in either the fitting of points to models or the assignment of models to points. However, the absence of cycles does guarantee that there are no impossible to resolve constraints, when stitching the parts together.

(body parts), the 3D coordinates of those points can be reconstructed using SfM approaches such as [64]. Similarly, if we knew the rigid motion parameters of each model (rotations and translations), segmentation could be easily performed by using the technique described in Chapter 5 to find which overlapping sets of points better fit the available models. This naturally suggests a hill climbing approach to the problem, where we by turn optimise model parameters and point assignment. Normally, the presence of many local optima is a concern with hill climbing approaches, as it makes such schemes highly dependent on the choice of initialisation. However, several recent works, including [77, 53], have shown that graph-cut based methods can be initialised with an excess of models making them much more robust to the choice of initialisation.

7.1.1 Assigning Points to Links

We consider a set of point tracks $\mathcal{P}$, and assume that tracks spatially adjacent to one another are connected in a graph structure. We express this by writing that each point track p is connected to a set of neighbours $\mathcal{N}_p$ (see section 7.1.3 for details on how the neighbourhood is built). This problem follows the formulation of the NRSfM problem described in Chapter 5: given this graph and a set of models $\mathcal{M}$, we choose an overlapping assignment of models to points $\mathbf{m} = \{m_1, m_2, \ldots, m_P\}$ by optimising the following cost function:

$$\underset{\mathbf{m}\in(2^{\mathcal{M}})^{\mathcal{P}}}{\arg\min}\, C(\mathbf{m}) = \sum_{p\in\mathcal{P}} \left(\sum_{\alpha\in m_p} U_p(\alpha) \right) + \mathrm{MDL}(\mathbf{m}), \tag{6.1}$$

subject to the constraints

$$\forall p \in \mathcal{P}\ \exists \alpha : p \in I_\alpha, \tag{5.2}$$

and

$$\forall q \in \mathcal{N}_p \wedge q \in I_\alpha \implies \alpha \in m_p. \tag{5.3}$$

where m_p is the subset of models assigned to point p and $U_p(\alpha)$ is the cost associated with assigning point p to model α (computed as the re-projection error defined in Equation 7.2). To avoid oversegmentation, we add a minimum description length prior [57, 29] $\mathrm{MDL}(\mathbf{m})$, which penalises the total number of active models[2] used to explain the data (for more details refer to Section 5.2.1).

The notation $p \in I_\alpha$ is short-hand for "p is an interior point of model α", where, as in topology, an interior point of a model or set α is defined as one whose neighbours must also belong to α. As such, constraint 5.3 defines an interior point; while constraint 5.2 states that every point must be an interior point of at least one model.

As discussed in Chapter 5, this differs from a conventional MRF formulation in that: firstly, while a particular point must belong to at least one model, it may belong to multiple models if it lies at the border between two models; and secondly, two neighbouring points in the graph must always share at least one model in common – this condition is enforced by constraints (5.2 and 5.3). This condition that neighbouring points must share models functions as a smoothing constraint, eliminating outliers and encouraging the use of a single model to explain spatially coherent regions. To optimise this problem we use the NOM formulation presented in Section 5.2.

7.1.2 3D Reconstruction of Rigid Segments

As is common on the A-SfM framework, we will use an orthographic camera model, a good mathematical approximation of the imaging process when the relief of the object is small considered to its distance to the camera — a valid assumption in the case of the human body (see Chapter 2). The problem of reconstructing a rigid object from an orthographic image stream was reviewed in Chapter 2. We now summarise the most important steps, and refer the reader to Section 2.1 for more details.

The 2D coordinates of a rigidly moving object $\mathtt{S}_r$ viewed by an orthographic cam-

[2]We use the term active model, to refer to a model which has at least one point belonging to it.

era are defined by:

$$\mathtt{W}_i = \mathtt{\Pi R}_i \mathtt{S}_r + \mathtt{T}_i, \tag{2.3}$$

where $\mathtt{\Pi}$ is the 2×3 orthographic projection matrix, $\mathtt{R}_i$ is a 3×3 rotation matrix (i.e. $\mathtt{R}_i \mathtt{R}_i^T = \mathtt{R}_i^T \mathtt{R}_i = \mathtt{I}_{3\times3}$) and $\mathtt{T}_i$ is a $2 \times P$ matrix describing the image translation. In the case of full data, if the image coordinates $\mathtt{W}_i$ are registered to the image centroid, the translation can be eliminated, resulting in $\tilde{\mathtt{W}}_i = \mathtt{W}_i - \mathtt{T}_i$. Stacking the registered image coordinates of all P points in all F frames gives the registered measurement matrix

$$\tilde{\mathtt{W}} = \begin{bmatrix} \tilde{\mathtt{W}}_1 \\ \tilde{\mathtt{W}}_2 \\ \vdots \\ \tilde{\mathtt{W}}_F \end{bmatrix} = \begin{bmatrix} \mathtt{\Pi R}_1 \\ \mathtt{\Pi R}_2 \\ \vdots \\ \mathtt{\Pi R}_F \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \cdots & \mathbf{s}_P \end{bmatrix} = \mathtt{MS}_r, \tag{2.4}$$

Estimating the model parameters $\mathtt{R}$ and $\mathtt{S}_r$ for each rigid segment can be formulated as the factorization problem [93] which minimizes image re-projection error:

$$\underset{\mathtt{R},\mathtt{S}_r}{\arg\min} \sum_{i=1}^{F} \left|\left|\tilde{\mathtt{W}} - \mathtt{\Pi R S}_r\right|\right|^2 \quad s.t. \;\; \mathtt{R}_i \mathtt{R}_i^T = \mathtt{R}_i^T \mathtt{R}_i = \mathtt{I}_{3\times3}. \tag{7.1}$$

In this thesis, instead of using the classical solution to factorization [93], we solve the problem via the Bundle Adjustment [99] non-linear optimisation approach (for more details see Section 2.2.1). We initialise our Bundle Adjustment formulation using the solution of Marques and Costeira [64] which has the advantage of providing rotation matrices that are guaranteed to lie on the manifold of matrices with orthonormal rows, and allows us to deal with missing data.

As was done in Equation 5.11 for the QD model on the NRSfM case, for the A-SfM

problem we define $U_p(\alpha)$ as the re-projection error of a rigid model α:

$$\begin{aligned} U_p(\alpha) &= \sum_{i=1}^{F}\sum_{p=1}^{P} \mathcal{R}(\mathbf{w}_{ip}, \mathbf{q}_i^\alpha, \mathbf{t}_i^\alpha, \mathbf{s}_p^\alpha) \\ &= \sum_{i=1}^{F}\sum_{p=1}^{P} \left|\left|\mathbf{w}_{ip} - \Pi\mathtt{R}_i^\alpha(\mathbf{q}_i^\alpha)\mathbf{s}_p^\alpha - \mathbf{t}_i^\alpha\right|\right|^2, \end{aligned} \tag{7.2}$$

where $\mathbf{q}_i^\alpha$ is the quaternion 4-vector parametrizing the 3×3 rotation matrix $\mathtt{R}_i^\alpha$ at frame i, $\mathbf{s}_p^\alpha = [X_p Y_p Z_p]^T$ are the the 3D coordinates of points p in a local referential, and $\mathbf{t}_i^\alpha$ is the 2-vector containing the translation component at frame i.

As discussed in Section 4.4.1, when performing piecewise reconstruction the global 3D object is recovered by aligning the shared points between segments and imposing the constraint that they must have the same 3D coordinates. This step is recurrent in piecewise reconstruction methods [34, 77, 90, 100]. In this case, the heuristic algorithm presented in Section 4.4.1 provided satisfactory results, for which it was chosen over the more complex algorithm from Section 6.3.1.

7.1.3 Guaranteeing a Valid Reconstruction

Our approach has two requirements, *(i)* to perform a reconstruction at least 3 points must belong to each active model, and *(ii)* to reduce the sign, or depth ambiguity, to a single binary decision per skeletal structure, models must be path connected by overlapping regions *i.e.* if model A intersects with B and B intersects with C, there is only one sign ambiguity to resolve for the entirety of A, B, and C.

Both of these properties are guaranteed by our inference approach. Property *(i)* holds for any neighbourhood structure in which every point has at least two neighbours. For a model α to be active, it must be an interior model of at least one point, *i.e.* $\alpha \in I_p$. If this point p is neighbours with at least two points q and r then $\alpha \in m_q$ and $\alpha \in m_r$ by constraint (5.3) $\square$.

Property *(ii)* holds providing the underlying neighbourhood structure is path con-

nected. This is a consequence of the fact that, if two points p_1 and p_n are path connected by the sequence $\{p_1, p_2, \ldots, p_n\}$, the models α_1 and α_n are also path connected by the sequence of models $\{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ where $\alpha_k \in I_{p_k}$. This last statement holds as constraint (5.3) guarantees that the interior models of neighbouring points must overlap $\square$. Neither of these properties need hold in a conventional MRF, such as those used by [53] where each point only belongs to exactly one model, and an active model may only have one point assigned to it.

Choice of neighbourhood structure

The neighbourhood structure used by our algorithm depends on both the distance measure chosen to tell how far apart points are, and a graph-building technique such as k-nearest neighbours, or minimum spanning tree. As we only want a plausible neighbourhood, and are uninterested in the physical or geodesic distance between points, we take the distance between points $\mathbf{x}_t$ and $\mathbf{y}_t$ in frame t as:

$$d_t(\mathbf{x}_t, \mathbf{y}_t) = w_1||\mathbf{x}_t - \mathbf{y}_t|| + w_2||\dot{\mathbf{x}}_t - \dot{\mathbf{y}}_t|| \tag{7.3}$$

i.e. as a weighted average of velocity and image distances. We take the final distance $d(\mathbf{x}, \mathbf{y})$ over all frames as the median of the 5% of greatest distances $d_t(\mathbf{x}_t, \mathbf{y}_t)$ divided by the number of frames both tracks occur in simultaneously.

Our choice of measure is robust to outliers, and separates tracks that (*a)* are spatially distinct; (*b)* move with different velocity; or (*c)* rarely occur in common frames.

To guarantee that properties *(i)* and *(ii)* hold, some care must be taken when choosing the neighbourhood structure of the graph. For example, the use of k-nearest neighbours where $k \geq 2$ would guarantee property *(i)*, while use of a minimum spanning tree would guarantee property *(ii)*. There seems to be no standard method that guarantees both required properties, and does not lead to an over-connect graph, so in practice we use 6-nearest neighbour as an initialisation and add to it additional minimum cost

edges, until we force the existence of at least two paths that do not share edges between every pair of points.

The existence of such paths connecting all points could potentially create problems in the reconstruction. For instanece, if our scene consists of different objects, these new paths would join them into the same neighbourhood structure. Given that our method relies on overlapping patches, these neighbourhood edges could force different objects to overlap, at the expense of the quality of the 3D reconstructions. Creating such edges can be avoided by only connecting points where the egde length is above a certain treshold, which coud be adaptively estimated from the data (e.g. the maxium allowed egde length for the additional paths could be the average edge length created by the 6-nearest neighbour connections plus 3 standard deviations). When this results in disjoint sets of points, we treat each of the sets independently, and continue to create the two paths between all pairs of points in each set. However, it is possible that such treshold is not enough to separate different objects in the neighbourhood structure. In such cases, we rely on the objects having different motion, such that the outlier rejection step would be able to correctly 'break' such connections and not create the overlap.

In pracitce, the addition of paths connecting every point was shown to be a perform well in the recovery from sparse point track data where the 6-nearest neighbour edges were not enough to correcly connect points belonging to the same articulated structure, while our smoothness constraints and outlier detection step prooved suitable in separating the few undesirable connections that appeared in our tests.

Initialisation

To initialise our approach we must propose a set of possible labels and corresponding model parameters to each of the $\mathcal{P}$ points. To avoid becoming stuck in a bad local optimum, we initialise with an excess of models, choosing the initial set $\mathcal{M}$ by fitting one model to each point $p \in \mathcal{P}$, and all of its neighbours. Given these initial labels the

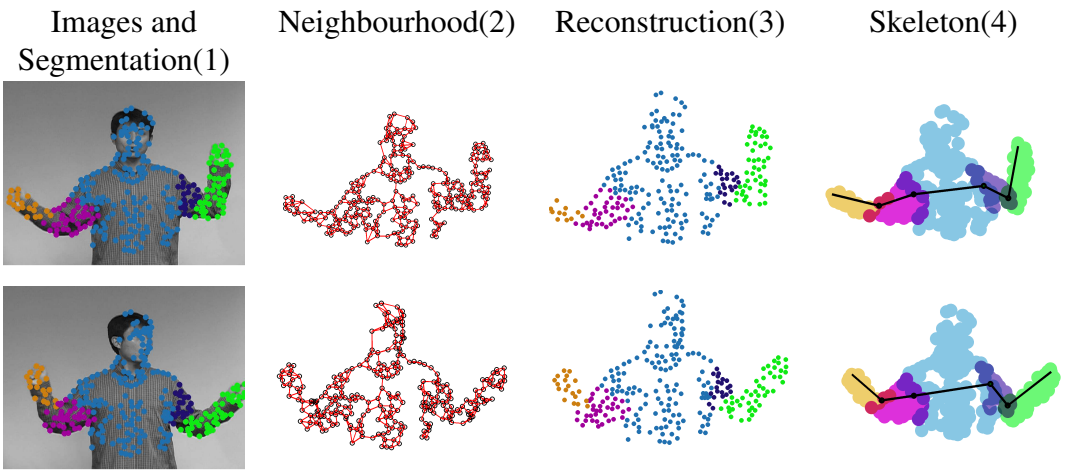


Figure 7.2: Reconstruction results from the **Dance** dataset [106]. **From left to right** (1) Original image and point location and decomposition). (2) Generated neighbourhood structure using the technique described in section 7.1.3. (3) Resulting decomposition into rigid overlapping models and estimated 3D reconstructions. (4) Estimated skeletal structure, and model assignment. Note that each node represents an intersection between two rigid models, and each edge the connecting model between two points. The location of the nodes is found by averaging all points which lie in the intersection.

initial model parameters are recovered by using the factorization approach of Marques and Costeira [64].

7.1.4 Missing Data and Multiple Articulated Objects

Neither the graph-cut based inference of section 7.1.1, nor the reconstruction algorithm of 7.1.2 requires complete point tracks, and can be applied to partial tracks. The only difficulty with the use of partial tracks is the generation of their neighbourhood structure. As some points are only visible for a short period of time, they may well be linked to the wrong section of the body, for example, points on the arm may be mistakenly linked to those on the torso, and while this may give a good reconstruction for the frames in which the points are visible, in other frames it can leave artefacts. To avoid these difficulties, we include points with more than 30% missing data directly in our framework, but give them an *empty neighbourhood*. Because of the MDL prior, these points without neighbours will belong to a common model used elsewhere in

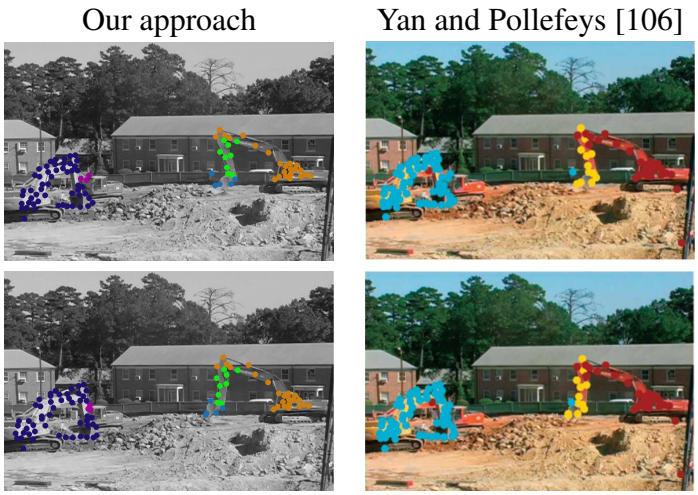


Figure 7.3: Two frames showing a comparison of our approach vs [106] on the digger dataset [106]. Compared to [106] we successfully segment the third digger at the back (magenta points), and decompose the right most digger into 3 components rather than the two found by [106]. See discussion in section 7.2.

the reconstruction. The procedure is equivalent to assigning partial tracks to the active model which minimises the re-projection error.

7.2 Experimental Results

We evaluate our approach on some of the more challenging articulated sequences in the literature. First, against the **Dance** (Figures 2.13 and 7.2), **Digger** (Figure 7.3), and **Toy** (Figure 7.4) sequences from [106], and further on the **Marple 13** sequence from [20] (Figure 7.7), the **Cat** sequence from [73] (Figure 7.5) and the **Skin** sequence of [71] (Figure 7.6), which has 3 dimensional ground truth. Despite the relatively poor quality, and under connected neighbourhoods (in both human cases, the torso can be separated into two sections linked only by a single point); the neighbour structure is sufficient to guarantee properties *(i)*, and *(ii)* of section 7.1.3, and the decomposition into models and final reconstruction is convincing. Compared to [106], our assignment of points to models is much smoother, with no outliers. This can be attributed to our requirement that adjacent models must overlap, which functions as a smoothing term,

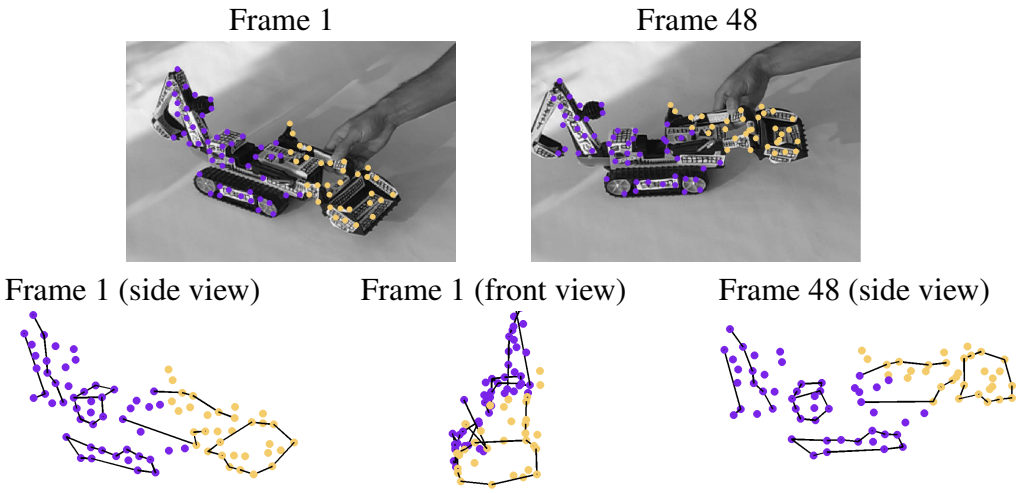


Figure 7.4: Decomposition and reconstruction of the **Toy** sequence from [106]. The two parts which move separately are successfully identified, while the 'tail' of the digger which is static with respect to the movement of the main body remains unseparated. Lines are added to improve visualization of results. See section 7.2 for more details.

suppressing outliers. Our failure to identify the head, as a model separate from the body in the **Dance** dataset (Figure 7.2) can be attributed to the same smoothing. In this sequence, the points on top of the head are incorrectly tracked, and [106] labels them as belonging to the torso (see Figure 2.13). With few points belonging to the head, and the points surrounding it belonging to the torso, its segmentation is suppressed.

We perform substantially better than [106] on the **Digger** dataset (Figure 7.3), showing our approach to be both more robust to outliers (*c.f.* blue point bottom row, far left), and more discriminative, as we both detect the motion of the bucket on the rightmost digger, and successfully separate movement of the third digger in the background. In this sequence, we followed [106] in thresholding the size of connecting edges – allowing multiple disconnected objects.

The dataset from [20] provides point tracks in several sequences of shots from detective stories. In Figure 7.7 we show qualitative results of our method on the **Marple 13** sequence using the provided tracks as input. Background tracks were removed using the segmentation results from [20]. Our method is once again able to provide

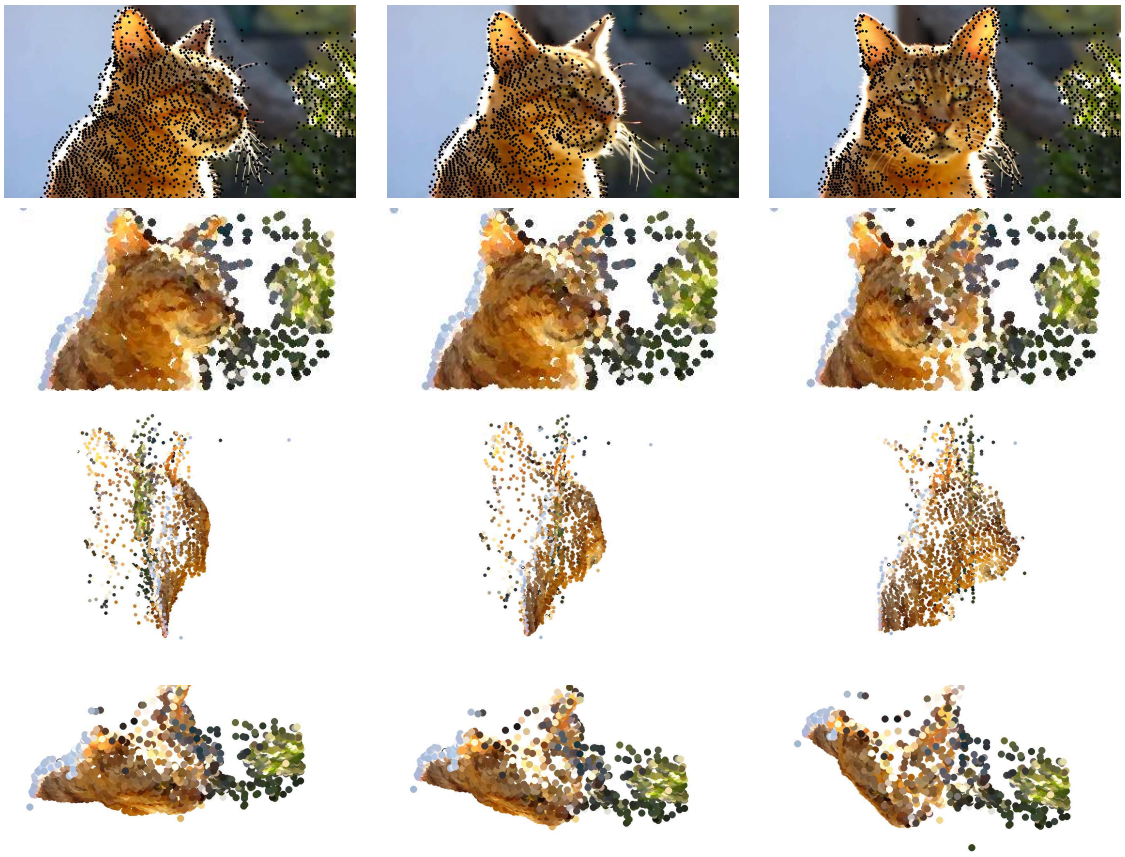


Figure 7.5: First row: Tracks for the **Cat** sequence provided by [73] – tracks estimated with [20]. Second, third and fourth rows: respectively the frontal, side and top views of the 3D reconstruction. Points are shown with the colour value of the first image when they are tracked.

reliable segmentation and reconstruction of the motion of the head, torso/neck, arm and forearm.

The **Cat** dataset from [73] is particularly challenging, as half of the head of the cat is occluded in the initial frames of the sequence. In addition, there are several points in the background that are stationary, which create outliers in the tracks. We show our reconstruction results in Figure 7.5, where the head of the cat is fully reconstructed and correctly merged into its body. The background as merged to the body of the cat as there are several static points in both cases and so it is impossible to segment them.

The **Skin** dataset from [71] was acquired using a Motion Capture setup consisting of 12 infra-red cameras tracking the 3D positions of approximately 350 reflective

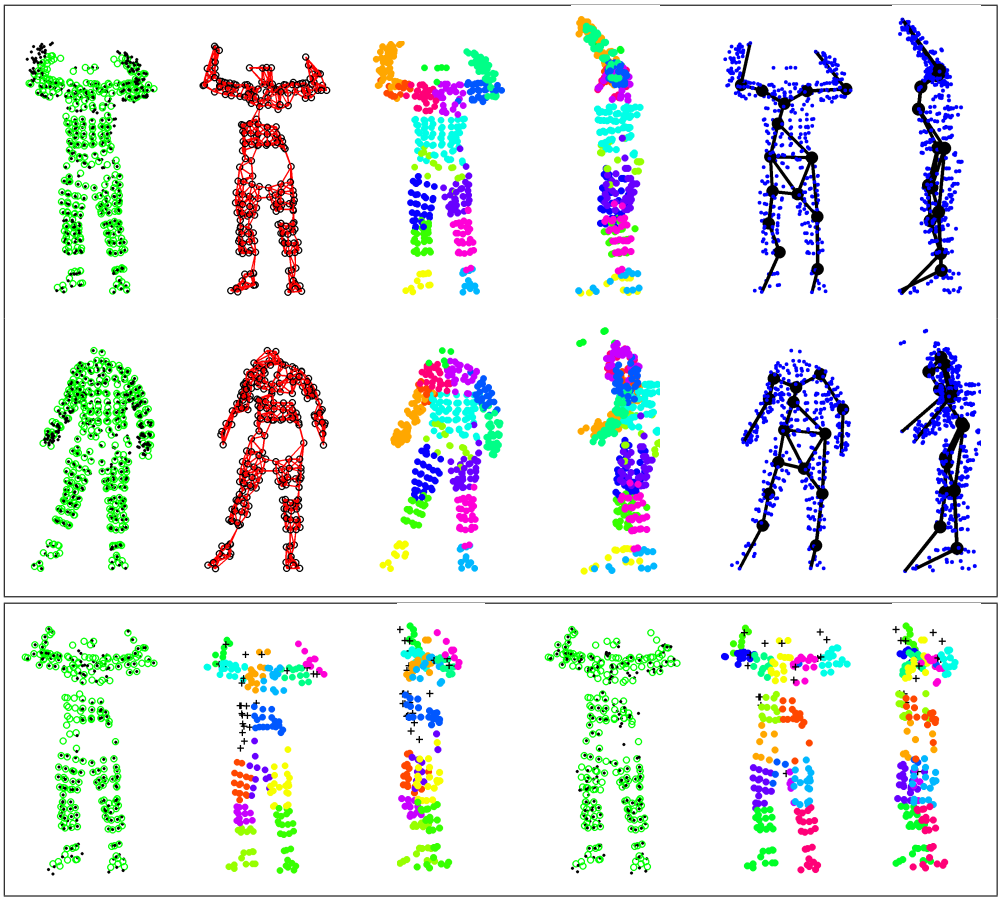


Figure 7.6: **Top** Our segmentation and reconstruction results on the dataset [71]. (a) Comparison between original ground truth (green) and 3D reconstruction (black) from a novel view point. (b) Generated neighbourhood structure using the technique described in section 7.1.3. (c) Resulting decomposition into rigid overlapping models and estimated 3D reconstructions. (d) Estimated skeletal structure from different view points. Note that each node represents an intersection between two rigid models, and each edge the connecting model between two points. **Bottom** Reconstruction and segmentation results using [106]. (e) and (g) Comparison between ground truth (green) and 3D reconstruction (black) using 14 and 15 segments respectively. (f) and (h) Segmentation results shown on GT data, with points discarded by RANSAC represented as black crosses.

markers – resulting in 467 tracks, some full and some partial. We project the 3D sequence using an orthographic camera model, and use our method to recover the 3D coordinates. Our results are shown on the first two rows of Figure 7.6. We use all tracks, full and partial. Measuring the error as the Frobenious norm of the difference between ground truth and reconstructed 3D points, divided by the Frobenious norm of the ground truth, we obtain a mean reconstruction error of 7.13% on this sequence.

Crucially, our algorithm does not misclassify any points, therefore there are no outliers to remove via RANSAC. Ignoring the hands (see Figure 7.6 column 1), which leaves 357 tracks, the error falls to 4.87%.

The third row of Figure 7.6 shows the segmentation of [106]. As only their spectral clustering code was available and not their automatic rank detection, we tested multiple parameters for the rank, number of neighbours, and number of segments and chose those that provided best results. RANSAC was then performed on the segmentation results to remove outliers. To measure the error we aligned the reconstruction of each segment with the corresponding ground truth points, bypassing the estimation of the kinematic chain as no code was available. This method can only use full tracks.

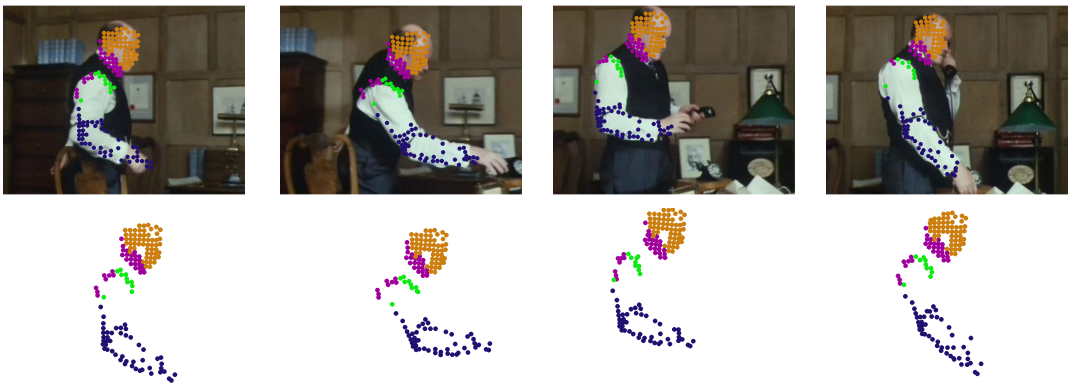


Figure 7.7: Segmentation and 3D reconstruction results from the data-set *Marple 13* [20] using the provided tracks.

In Figure 7.6 (third row) we show two of the best results achieved with [106]. Using rank 6, 14 neighbour points and 14 segments (Figure 7.6, (e) and (f)) 195 out of the 219 full tracks available were reconstructed with a reconstruction error of 5.09% – over all 219 tracks the error was 6.15%. Note that the resulting segmentation has the right knee as an extra object, combines the left foot with the left lower leg, and merges the inner region of both legs. Using rank 6, 18 neighbours and 15 segments (Figure 7.6, (g) and (h)) [106] reconstruct 204 out of the 219 points, (15 points were rejected by RANSAC), with a reconstruction error of 6.96% or 8.01% over the complete tracks. This segmentation also combines part of both legs as one object, merges each foot

with its corresponding lower leg and treats the right elbow as a new object. As we aligned each segment obtained with [106] with ground truth, their error measures are artificially low and relied upon knowledge of the true 3D positions.

7.3 Conclusion

In this chapter we have shown how the NOM formulation presented in Chapter 5 can be used in a data-driven approach for the problem of simultaneous segmentation and 3D reconstruction of articulated motion. Without any assumptions about the skeletal structure of the object we reconstruct, we are able to obtain both high quality 3D reconstructions, and a semantically meaningful decomposition into articulated parts. Compared to existing motion segmentation approaches, we strongly benefit from spatial smoothing priors, which both increase our robustness to outliers, and make it easier for us to recover semantically informative segmentations.

We improve substantially on previous articulated SfM methods which were only demonstrated on simple two part articulated sequences with full data, by demonstrating our complete system on challenging full body human articulated sequences and providing a principled solution to dealing with missing data.

We performed 3D reconstructions on a range of real sequences where we compared qualitatively with existing methods for articulated motion reconstruction. These experiments showed how versatile our approach is, reconstructing not only human motion but also other articulated objects such as construction diggers. Additionally we show qualitative results on two realistic sequences, where a significant amount of missing tracks and outliers are present.

Quantitative analysis is performed on a full human body motion sequence where ground truth was provided by a MoCap system. Our results showed improved 3D reconstruction performance over the state of the art in addition to a more plausible segmentation of the rigid parts.

A summary of our proposed methods is presented in Table 7.1.

Table 7.1: Summary of presented algorithms.

Algorithm	**Piecewise**	**Model**	**Adaptive**	**Initialization**	**Missing Data**
Quad (Chapter 3)	No	QD	No	Rigid SfM (from first few frames)	Can lose tracks in $\mathtt{S}_q$ Cannot incorporate new tracks
Piecewise-Quad (Chapter 4)	Yes	QD	No	Rigid SfM (+ Isomap if known to be flat)	Can lose tracks in $\mathtt{S}_q$ (per patch) Cannot incorporate new tracks
NOM+Piecewise-Quad (Chapter 5)	Yes	QD (supports multiple types)	Yes	Rigid SfM (+ Isomap if known to be flat)	Can lose tracks in $\mathtt{S}_q$ (per patch) Cannot incorporate new tracks
NOM+Piecewise-Rigid (Chapter 7)	Yes	Rigid	Yes	Rigid SfM	Can lose and incorporate new tracks

Chapter 8

Conclusions

This thesis tackled the problem of non-rigid structure from motion (NRSfM): recovering the 3D geometry of a deformable scene observed by a single moving camera. In particular, we focus on the case where the observed scene consists of an object with strong local deformations, such as a flag waving in the wind, and studied the limitations of state of the art methods in such scenarios. We argue that methods that model highly deformable objects *globally* fail to reconstruct such scenes due to the high complexity of the observed motion. In particular, methods based on the low-rank basis shape model of Bregler *et al*. [18], which have dominated the NRSfM literature in the last decade, overfit to the data due to the high number of bases need to deal with such complex deformations.

As part of a recent trend in the NRSfM community, we argue that reconstructing such complex deformations is a problem better solved by modelling objects *locally*. These methods [100, 90, 22], like the solutions we propose in Chapters 4 to 6, perform 3D reconstruction in a *piecewise* fashion where each local region is reconstructed independently and later merged into the global object reconstruction.

Typically, piecewise methods divide the scene into local regions by requiring manual input [100] or relying on the chosen local model to provide a implicit division such as Taylor *et al*.'s *triangle soup* approach [90]. Instead, we show how the division into

regions (or patches) and local reconstruction can be formulated in a *principled* way. We formulate the patch division and local reconstruction problem as an alternating approach, where the same geometric cost – the image re-projection error of the 3D reconstruction – is minimised. This is possible by formulating the patch division problem as a labelling problem, with the additional requirement that neighbouring patches must overlap. This is needed to provide cues to merge the individual patches in 3D, which is done by enforcing consistency between the reconstructions of overlapping regions.

In addition to this framework, we provide our own local reconstruction model – the Quadratic Deformation (QD) model – and support our choice with a set of experiments on synthetic and real data, comparing to benchmark methods based on the low-rank shape basis model and other piecewise approaches [100, 90]. Finally, we show how the reconstruction with this model can be scaled to *dense* data, where instead of reconstructing a set of sparse feature point tracks we work directly on multi-frame optic flow [42, 92].

Our piecewise approaches to non-rigid reconstruction proposed in Chapters 3 to 6 can easily deal with points that go out of view throughout a sequence by only considering the costs where image data is available in the non-linear least-squares optimisation.

However, incorporating new points into the reconstruction that were not initially in the neighbourhood structure requires future improvements. When considering the dense NRSfM approach described in Chapter 6, the observation that allows for the pre-computation of the shape and image factors also limits us to reconstruct only the points that are visible in the first frame. When that is not the case, the number of tracks is different at every frame, meaning a different shape and image factor is needed, thus preventing their pre-computation.

This limitation has forced us to work on relatively short sequences since most tracking methods drift over a long period of time. Moreover, situations where initially occluded parts of the object might become visible, due either to external or self-

occlusions or where a new object comes into view are frequent in real life sequences. It is thus desirable to account for new tracks in our formulation, as it would add robustness and increase the applicability of our NRSfM solution to more challenging sequences.

While our approach to 3D reconstruction of articulated structure (see Chapter 7) addresses this problem by incorporating a strategy to add new points into the neighbourhood structure to allow their reconstruction and we have provided experimental evaluations on a challenging real-life sequence of a cat, the non-rigid case requires more careful attention. Future work will address this problem by analysing how new tracks can be incorporated into existing models. The image location of these new tracks in the frame they first appear is a strong cue as to which existing model is should belong. After assigning these new tracks to existing models, we have two options to deal with the reconstruction problem in the current framework: either we compute different sets of transformations relative to different images for every model, which is unlikely to be efficient; or we use the model parameters based on known tracks to compute a reverse warp of new tracks towards our reference image, enabling us to keep referring the model parameters to the same image, which results in a more efficient solution. Computing a reverse-warp is not trivial, and thus other constraints to help solve the problem need to be investigated.

Contemporary NRSfM methods rely on previously computed point tracks or optic flow, and assume tracking to be an independent problem. However, it is common in the tracking community to use motion subspace constraints [52, 97, 42] to provide better estimates for feature tracks or multi-frame optic flow. Similarly to Brand's approach for the low-rank shape basis model [17], reconstruction and tracking could be integrated into the same framework. Following the success of our simultaneous patch division and reconstruction, it would be advantageous to include the tracking step in the same optimisation, where the tracking of points in the image sequence is guided by the 3D non-rigid geometry of the scene, and vice-versa. In our approach, we have

shown how the key to solve this problem is the formulation of a common geometric cost for both the patch division and reconstruction step. Further work in this direction would require the design of common geometric cost that would unify piecewise tracking, segmentation and reconstruction which would allow to perform 3D reconstruction directly from the raw video instead of from point matches.

Our principled formulation for simultaneous segmentation and reconstruction can also be linked to Malik's 'Recognition, Reconstruction and Reorganisation' paradigm for vision [62]. In this light, our approach can be seen as performing simultaneous reconstruction and reorganisation by optimising a single geometric cost to solve both problems. It then becomes clear that our approach is lacking the recognition step. An interesting thread for future research is how to perform recognition based on the recovered 3D shape, motion and scene segmentation. For instance, when applying the approach described in Chapter 7, the set of segments and the underlying 3D skeleton can be used as a cue to recognise the type of object being observed. On the other hand, knowledge about the class of object being observed could constrain the reconstruction and segmentation process. As an example, when dealing with human motion, such knowledge could be used to guide the segmentation process to provide an articulated tree that matches a prior model of human articulated skeleton. This knowledge could potentially provide information to resolve ambiguities and further refine the reconstruction process, leading to better overall results in all three steps.

Bibliography

[1] Henrik Aanæs and Fredrik Kahl. Estimation of deformable structure and motion. In *Workshop on Vision and Modelling of Dynamic Scenes, Copenhagen, Denmark*, 2002. 18, 40, 72

[2] Ankur Agarwal and Bill Triggs. 3d human pose from silhouettes by relevance vector regression. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Washington D.C.*, pages 882–888, 2004. 57

[3] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, and Richard Szeliski. Dense 3d motion capture from synchronized video streams. *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*, 2008. 21

[4] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M. Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011. 16, 17

[5] Ijaz Akhter, Yaser Sheikh, and Sohaib Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*, pages 1534–1541, 2009. 36, 46, 73

[6] Ijaz Akhter, Yaser Ajmal Sheikh, Sohaib Khan, and Takeo Kanade. Nonrigid structure from motion in trajectory space. In *Neural Information Processing Systems*, December 2008. 18, 41, 42, 43, 44, 47, 56, 71

[7] Adrien Bartoli, Vincent Gay-Bellile, Umberto Castellani, Julien Peyras, Søren Olsen, and Patrick Sayd. Coarse-to-Fine Low-Rank Structure-from-Motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*, 2008. 18, 40, 60

[8] Adrien Bartoli, Yan Gérard, François Chadebecq, and Toby Collins. On template-based reconstruction from a single view: Analytical solutions and proofs of well-posedness for developable, isometric and conformal surfaces. *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Providence, Rhode Island*, June 2012. 24, 53, 55

[9] Bernd Bickel, Mario Botsch, Roland Angst, Wojciech Matusik, Miguel Otaduy, Hanspeter Pfister, and Markus Gross. Multi-scale capture of facial geometry and motion. *Proceedings of ACM SIGGRAPH, San Diego*, 26(3):33.1–33, August 2007. 21, 22

[10] Michael Bleyer, Carsten Rother, and Pushmeet Kohli. Surface stereo with soft segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, California*, 2010. 105

[11] Endre Boros and Peter L. Hammer. Pseudo-boolean optimization. *Discrete Applied Mathematics*, page 155 225, 2002. 136

[12] Endre Boros, Peter L. Hammer, and Gabriel Tavares. Preprocessing of unconstrained quadratic binary optimization. Technical Report RRR 10-2006, RUTCOR, Apr 2006. 136

[13] boujou. 2d3 Ltd. `http://www.2d3.com`, 2007. 16, 17

[14] Yuri Boykov and Vladimir Kolmogorov. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004. 130

[15] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23, 2001. 108, 130

[16] Matthew Brand. Morphable 3d models from video. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*, volume 2, pages 456–463, December 2001. 35, 123, 124

[17] Matthew Brand. A direct method for 3D factorization of nonrigid motion observed in 2D. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, pages 122–128, 2005. 35, 60, 71, 168

[18] Christoph Bregler, Aaron Hertzmann, and Henning Biermann. Recovering non-rigid 3D shape from image streams. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina*, pages 690–696, June 2000. 17, 18, 28, 31, 32, 34, 41, 45, 47, 60, 124, 130, 166

[19] Christoph Bregler and Jitendra Malik. Tracking people with twists and exponential maps. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara*, 1998. 57, 147

[20] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 158, 159, 160, 162

[21] Florent Brunet, Richard Hartley, Adrien Bartoli, Nassir Navab, and Remy Malgouyres. Monocular template-based reconstruction of smooth and inextensible

surfaces. In *Proc. 10th Asian Conference on Computer Vision, Queenstown, New Zeland*, 2010. 53, 88, 124, 130

[22] Toby Collins and Adrien Bartoli. Locally affine and planar deformable surface reconstruction from video. *VMV'10 - Proceedings of the International Workshop on Vision, Modeling and Visualization*, 11/2010 2010. 18, 19, 24, 51, 52, 82, 148, 166

[23] J. P. Costeira and T. Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3):159 – 179, 1998. 16, 57

[24] Yuchao Dai, Li Hongdong, and Mingyi He. A simple prior-free method for non-rigid structure-from-motion factorization. *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Providence, Rhode Island*, June 2012. 46, 47, 56

[25] Alessio Del Bue and Lourdes Agapito. Stereo non-rigid factorization. *International Journal of Computer Vision*, 66(2):193–207, February 2006. 18, 40, 80

[26] Alessio Del Bue, Fabrizio Smeraldi, and Lourdes Agapito. Non-rigid structure from motion using ranklet–based tracking and non-linear optimization. *Image and Vision Computing*, 25(3):297–310, March 2007. 18, 75, 83, 95, 98

[27] Alessio Del Bue, João Xavier, Lourdes Agapito, and Marco Paladini. Bilinear factorization via augmented lagrange multipliers. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, volume 6314, pages 283–296. Springer, 2010. 38

[28] Andrew Delong, Anton Osokin, Hossam Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, California*, 2010. 127

[29] Andrew Delong, Anton Osokin, Hossam N. Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, 2010. 105, 111, 113, 120, 152

[30] Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Soc.*, 39(1), 1977. Series B. 104, 112, 113

[31] Piotr Dollár, Vincent Rabaud, and Serge Belongie. Non-isometric manifold learning: Analysis and an algorithm. In *24th International Conference in Machine Learning, Corvallis, OR, USA*, 2007. 44

[32] Ehsan Elhamifar and René Vidal. Sparse subspace clustering. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*, 2009. 150

[33] Olivier D. Faugeras, Quang-Tuan Luong, and Stephen Maybank. Camera self-calibration: Theory and experiments. In *Proc. 2nd European Conference on Computer Vision, Santa Margharita Ligure, Italy*, LNCS 588, pages 321–334, 1992. 16

[34] João Fayad, Lourdes Agapito, and Alessio Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 18, 19, 24, 25, 41, 82, 119, 139, 148, 149, 154

[35] João Fayad, Alessio Del Bue, Lourdes Agapito, and Pedro M.Q. Aguiar. Human body modelling using quadratic deformations. In *7th EUROMECH Solid Mechanics Conference, Lisbon, Portugal*, 2009. 61

[36] João Fayad, Alessio Del Bue, Lourdes Agapito, and Pedro M.Q. Aguiar. Non-rigid structure from motion using quadratic deformation models. In *Proc. 20th British Machine Vision Conference, London*, 2009. 18, 24, 98, 99, 119

[37] João Fayad, Chris Russell, and Lourdes Agapito. Automated articulated structure and 3d shape recovery from point correspondences. In *Proc. 13th International Conference on Computer Vision, Barcelona, Spain*, 2011. 26, 123

[38] Pedro F. Felzenszwalb and Daniel Huttenlocher. Distance transforms of sampled functions. Technical Report TR2004-1963, Cornell University, 2004. 139

[39] David A. Forsyth, Okan Arikan, Leslie Ikemoto, James O'Brien, and Deva Ramanan. Computational studies of human motion: part 1, tracking and motion synthesis. *Found. Trends. Comput. Graph. Vis.*, July 2005. 56

[40] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *PAMI*, 32(8):1362–1376, 2010. 125, 126

[41] David Gallup, Jan-Michael Frahm, and Marc Pollefeys. Piecewise planar and non-planar stereo for urban scene reconstruction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, California*, 2010. 125, 126

[42] Ravi Garg, Anastasios Roussos, and Lourdes Agapito. Robust trajectory-space tv-l1 optical flow for non-rigid sequences. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2011. 139, 167, 168

[43] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. JHU Press, 1996. 65

[44] Paulo Gotardo and Aleix Martinez. Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):2051–2065, October 2011. 43, 44, 46, 56

[45] Paulo Gotardo and Aleix Martinez. Model evolution: An incremental approach to non-rigid structure from motion. *Proc. 13th International Conference on Computer Vision, Barcelona, Spain*, November 2011. 41, 45, 60

[46] Richard Hartley and René Vidal. Perspective nonrigid shape and motion recovery. In *Proc. 10th European Conference on Computer Vision, Marseille, France*, pages 276–289, 2008. 36

[47] Richard I. Hartley and Federik Schaffalitzky. Reconstruction from projections using grassmann tensors. In *Proc. 8th European Conference on Computer Vision, Prague, Czech Republic*, 2004. 36

[48] Richard I. Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 15

[49] Carlos Hernandez, George Vogiatzis, Gabriel J. Brostow, Bjorn Stenger, and Roberto Cipolla. Non-rigid photometric stereo with colored lights. In *Proc. 11th International Conference on Computer Vision, Rio de Janeiro, Brazil*, Rio de Janeiro, Brazil, October 2007. 21, 27

[50] Derek Hoiem, Carsten Rother, and John M. Winn. 3d layoutcrf for multi-view object class recognition and segmentation. In *CVPR*, 2007. 105, 111, 113

[51] Berthold K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America*, 4(4):629–642, 1987. 28

[52] Michal Irani. Multi-frame optical flow estimation using subspace constraints. In *Proc. 7th International Conference on Computer Vision, Kerkyra, Greece*, pages 626–633, 1999. 168

[53] Hossam N. Isack and Yuri Boykov. Energy-based geometric multi-model fitting. *International Journal of Computer Vision*, 97(2):123–147, 2012. 25, 104, 105, 151, 155

[54] Natasha Kholgade, Iain Matthews, and Yaser Sheikh. Content retargeting using parameter-parallel facial layers. *Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, August 2011. 21, 22

[55] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *PAMI*, 28(10), 2006. 108

[56] Vladimir Kolmogorov and Carsten Rother. Minimizing non-submodular functions with graph cuts: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2), 2007. 136

[57] Lubor Ladicky, Chris Russell, Pushmeet Kohli, and Philip Torr. Graph cut based inference with co-occurrence statistics. In *ECCV*, 2010. 105, 111, 127, 152

[58] Danial Lashkari and Polina Golland. Convex clustering with exemplar-based models. In *NIPS*, 2007. 104

[59] Victor Lempitsky, Carsten Rother, Stefan Roth, and Andrew Blake. Fusion moves for markov random field optimization. *PAMI*, 2010. 136

[60] Hugh C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981. 16

[61] David MacKay. Chapter 20. an example inference task: Clustering. In *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 1998. 104

[62] Jitendra Malik. Recognition, reconstruction and reorganization. In *Material for the International Computer Vision Summer School 2012*, 2012. 12, 169

[63] Abed Malti, Adrien Bartoli, and Toby Collins. Template-based conformal shape-from-motion-and-shading for laparoscopy. In *International Conference on Information Processing in Computer-Assisted Interventions (IPCAI'12)*, Pisa, Italy, 2012. 55

[64] Manuel Marques and João P. Costeira. Estimating 3D shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, 2008. 74, 75, 90, 151, 153, 157

[65] David Marr. *Vision*. Freeman, San Francisco, 1982. 12, 125, 131

[66] Matthias Müller, Bruno Heidelberger, Matthias Teschner, and Markus Gross. Meshless deformations based on shape matching. In *SIGGRAPH 2005*, volume 24, pages 471–478, New York, NY, USA, 2005. ACM. 60, 73, 84

[67] Richard Newcombe, Steven J. Lovegrove, and Andrew J. Davison. Dtam: Dense tracking and mapping in real-time. *Proc. 13th International Conference on Computer Vision, Barcelona, Spain*, November 2011. 16, 17, 123, 125

[68] Denis Oberkampf, Daniel F. DeMenthon, and Lary S. Davis. Iterative pose estimation using coplanar feature points. *Computer Vision and Image Understanding*, 63:495–511, 1996. 133

[69] Marco Paladini, Alessio Del Bue, Marko Stosic, Marija Dodig, João Xavier, and Lourdes Agapito. Factorization for non-rigid and articulated structure using metric projections. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*, pages 2898–2905, 2009. 18, 38, 43, 44, 47, 56, 75, 95, 98, 130

[70] Hyun Soo Park, Takaaki Shiratori, Ian Matthews, and Yaser Sheikh. 3d reconstruction of a moving point from a series of 2d projections. In *Proc. 11th European Conference on Computer Vision, Crete, Greece*, 2010. 42, 43, 56

[71] Sang Il Park and Jessica K. Hodgins. Capturing and animating skin deformation in human motion. In *SIGGRAPH 2006*, pages 881–889, New York, NY, USA, 2006. ACM. 61, 73, 84, 158, 160, 161

[72] Mathieu Perriollat, Richard I. Hartley, and Adrien E. Bartoli. Monocular template-based reconstruction of inextensible surfaces. In *Proc. 19th British Machine Vision Conference, Leeds*, 2008. 53, 88

[73] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Providence, Rhode Island*, 2012. 158, 160

[74] Vincent Rabaud and Serge Belongie. Re-thinking non-rigid structure from motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*, pages 1–8, 2008. 18, 41, 44, 45, 60

[75] Alex Rav-Acha, Pushmeet Kohli, Carsten Rother, and Andrew Fitzgibbon. Unwrap mosaics: A new representation for video editing. *ACM Transactions on Graphics (SIGGRAPH)*, 2008. 133

[76] David A. Ross, Daniel Tarlow, and Richard S. Zemel. Learning articulated structure and motion. *International Journal of Computer Vision*, 88(2):214–237, 2010. 58

[77] Chris Russell, João Fayad, and Lourdes Agapito. Energy based multiple model fitting for non rigid structure from motion. In *Proc. IEEE Conference on Com-*

puter Vision and Pattern Recognition, Colorado Springs, Colorado, 2011. 25, 104, 105, 121, 123, 127, 136, 148, 149, 151, 154

[78] Chris Russell, João Fayad, and Lourdes Agapito. Dense non-rigid structure from motion. In *Proceedings of 3D Digital Imaging, Modeling, Processing, Visualization and Transmission, Zurich, Switzerland*, October 2012. 25

[79] Mathieu Salzmann and Pascal Fua. Reconstructing sharply folding surfaces: A convex formulation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida*, 2009. 54

[80] Mathieu Salzmann, Richard Hartley, and Pascal Fua. Convex optimization for deformable surface 3-d tracking. In *IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007. 53, 54

[81] Mathieu Salzmann, Francesc Moreno-Noguer, Vincent Lepetit, and Pascal Fua. Closed-form solution to non-rigid 3d surface registration. In *Proc. 10th European Conference on Computer Vision, Marseille, France*, 2008. 53, 54

[82] Mathieu Salzmann, Julien Pilet, Slobodan Ilic, and Pascal Fua. Surface deformation models for nonrigid 3d shape recovery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(8):1481 – 1487, June 2007. 53, 88

[83] Mathieu Salzmann, Raquel Urtasun, and Pascal Fua. Local deformation models for monocular 3d shape recovery. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska*, 2008. 53, 54, 55, 88, 124

[84] Steve Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, New York, NY*, June 2006. 14, 125, 126

[85] Francesco Setti, Mariolino De Cecco, and Alessio Del Bue. A multi-view stereo system for articulated motion analysis. In *Proceedings of the Fifth International Conference on Computer Vision Theory and Applications (VISAPP 2010)*, volume 1, pages 367–372. INSTICC Press, May 2010. 100, 101, 117

[86] Cristian Sminchisescu and Bill Triggs. Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research*, 22(6), 2003. 57, 147

[87] Jos F. Sturm. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999. 38

[88] Peter Sturm and Bill Triggs. A factorization based algorithm for multi-image projective structure and motion. In *Proc. 4th European Conference on Computer Vision, Cambridge*, pages 709–720, April 1996. 16

[89] Camillo J. Taylor. Reconstruction of articulated objects from point correspondences in a single image. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head, South Carolina*, pages 677–685, June 2000. 19, 147

[90] Jonathan Taylor, Allan D. Jepson, and Kiriakos N. Kutulakos. Non-rigid structure from locally-rigid motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, California*, 2010. 18, 19, 24, 49, 50, 51, 59, 82, 116, 117, 118, 119, 120, 121, 123, 126, 127, 130, 136, 137, 139, 148, 149, 154, 166, 167

[91] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 2000. 89, 90

[92] Yuandong Tian and Srinivasa G. Narasimhan. A globally optimal data-driven approach for image distortion estimation. *International Journal of Computer Vision (IJCV)*, 2011. 139, 167

[93] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: A factorization approach. *International Journal of Computer Vision*, 9(2):137–154, 1992. 16, 28, 30, 31, 32, 37, 56, 57, 70, 89, 90, 148, 150, 153

[94] Philip H. S. Torr and David W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24, 1997. 57

[95] Lorenzo Torresani and Aaron Hertzmann. Automatic non-rigid 3d modeling from video. In *Proc. 8th European Conference on Computer Vision, Prague, Czech Republic*, pages 299–312, May 2004. 18

[96] Lorenzo Torresani, Aaron Hertzmann, and Christoph Bregler. Non-rigid structure-from-motion: Estimating shape and motion with hierarchical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 878–892, 2008. 18, 37, 43, 47, 60, 75, 79, 83, 95, 98, 124, 130

[97] Lorenzo Torresani, Danny Yang, Gene Alexander, and Christoph Bregler. Tracking and modeling non-rigid objects with rank constraints. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, Kauai, Hawaii*, 2001. 37, 38, 168

[98] Philip Tresadern and Ian Reid. Articulated structure from motion by factorization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, volume 2, pages 1110–1115, June 2005. 16, 56, 57, 58, 148

[99] Bill Triggs, Philip McLauchlan, Richard I. Hartley, and Andrew Fitzgibbon. Bundle adjustment – A modern synthesis. In W. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, LNCS, pages 298–375. Springer Verlag, 2000. 39, 40, 133, 153

[100] Aydin Varol, Mathieu Salzmann, Engin Tola, and Pascal Fua. Template-free monocular reconstruction of deformable surfaces. In *International Conference on Computer Vision*, Kyoto, Japan, 2009. 18, 19, 24, 48, 49, 51, 52, 59, 82, 99, 100, 102, 119, 120, 121, 127, 130, 139, 148, 149, 154, 166, 167

[101] René Vidal, Yi Ma, and Shankar Sastry. Generalized principal component analysis (gpca). In *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Diego, California*, pages 621–628, 2003. 57, 150

[102] Ryan White, Keenan Crane, and David Forsyth. Capturing and animating occluded cloth. In *ACM Tranactions on Graphics (SIGGRAPH)*, 2007. 83, 95, 96

[103] Robert J. Woodham. *Photometric method for determining surface orientation*. Optical Engineering, 1980. 27

[104] Jing Xiao, Jin-Xiang Chai, and Takeo Kanade. A closed-form solution to non-rigid shape and motion recovery. *International Journal of Computer Vision*, 67(2):233–246, April 2006. 18, 34, 35, 47, 60, 71

[105] Jing Xiao and Takeo Kanade. Uncalibrated perspective reconstruction of deformable structures. In *Proc. 10th International Conference on Computer Vision, Beijing, China*, October 2005. 75

[106] Jingyu Yan and Mark Pollefeys. A factorization-based approach for articulated non-rigid shape, motion and kinematic chain recovery from video. *IEEE Trans-*

actions on Pattern Analysis and Machine Intelligence, 30(5), May 2008. 16, 56, 57, 58, 148, 157, 158, 159, 161, 162, 163

[107] Lihi Zelnik-Manor and Michal Irani. Degeneracies, dependencies and their implications in multi-body and multi-sequence factorizations. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 287–293, June 2003. 57

[108] Huiyu Zhou and Huosheng Hu. Human motion tracking for rehabilitationa survey. *Biomedical Signal Processing and Control*, 3(1):1 – 18, 2008. 22

[109] Shengqi Zhu, Li Zhang, and Brandon M. Smith. Model evolution: An incremental approach to non-rigid structure from motion. *Proc. IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, California*, June 2012. 44, 45

Appendix A

Efficient Quadratic Surface fitting

We show how $C^{i,\alpha}(\mathtt{A}_i, \mathtt{R}_i, \mathbf{t}_i) = \sum_{j\in\alpha} ||\mathbf{w}_{ij} - \mathtt{R}_i\mathtt{A}_i\mathbf{s}_j - \mathtt{T}_i||_2^2$ (the aggregate cost for a singleframe i over all the points belonging to model α) can be efficiently calculated. For clarity, throughout this derivation we drop the index i. $\langle a, b\rangle$ is the inner product between two vectors a and b of the same size, and satisfies the properties:

$$\langle a, b\rangle = a^\top b = {}^\top(ab^\top), \qquad \langle a, a\rangle = ||a||_2^2, \qquad \text{and} \qquad \langle ac, b\rangle = \langle c, a^\top b\rangle. \quad \text{(A.1)}$$

Then,

$$||\mathbf{w}_j - \mathtt{RA}\mathbf{s}_j - \mathbf{t}||_2^2 \tag{A.2}$$

$$=\langle \mathbf{w}_j - \mathtt{RA}\mathbf{s}_j - \mathbf{t}, \mathbf{w}_j - \mathtt{RA}\mathbf{s}_j - \mathbf{t}\rangle \tag{A.3}$$

$$=||\mathbf{w}_j||_2^2 - 2\langle \mathbf{w}_j, \mathtt{RA}\mathbf{s}_j + \mathbf{t}\rangle + ||\mathtt{RA}\mathbf{s}_j + \mathbf{t}||_2^2 \tag{A.4}$$

$$=||\mathbf{w}_j||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2\langle \mathbf{w}_j, \mathtt{RA}\mathbf{s}_j\rangle + ||\mathtt{RA}\mathbf{s}_j + \mathbf{t}||_2^2 \tag{A.5}$$

$$=||\mathbf{w}_j||_2^2 + ||\mathtt{RA}\mathbf{s}_j||_2^2 + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2\langle \mathbf{w}_j, \mathtt{RA}\mathbf{s}_j\rangle + 2\langle \mathtt{RA}\mathbf{s}_j, \mathbf{t}\rangle \tag{A.6}$$

$$=||\mathbf{w}_j||_2^2 + ||\mathtt{RA}\mathbf{s}_j||_2^2 + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2\langle \mathbf{w}_j, \mathtt{RA}\mathbf{s}_j\rangle + 2\langle \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle \tag{A.7}$$

$$=||\mathbf{w}_j||_2^2 + ||\mathtt{RA}\mathbf{s}_j||_2^2 + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2^\top(\mathbf{w}_j(\mathtt{RA}\mathbf{s}_j)^\top) + 2\langle \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle \tag{A.8}$$

$$=||\mathbf{w}_j||_2^2 + ||\mathtt{RA}\mathbf{s}_j||_2^2 + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2^\top(\mathbf{w}_j\mathbf{s}_j{}^\top(\mathtt{RA})^\top) + 2\langle \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle \tag{A.9}$$

$$=||\mathbf{w}_j||_2^2 + {}^\top(\mathtt{RA}\mathbf{s}_j(\mathtt{RA}\mathbf{s}_j)^\top) + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2^\top(\mathbf{w}_j\mathbf{s}_j{}^\top(\mathtt{RA})^\top) + 2\langle \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle \tag{A.10}$$

$$=||\mathbf{w}_j||_2^2 + {}^\top(\mathtt{RA}\mathbf{s}_j\mathbf{s}_j{}^\top(\mathtt{RA})^\top) + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2^\top(\mathbf{w}_j s_j^\top(\mathtt{RA})^\top) + 2\langle \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle \tag{A.11}$$

$$=||\mathbf{w}_j||_2^2 + {}^\top(\mathtt{RA}\mathbf{s}_j\mathbf{s}_j{}^\top(\mathtt{RA})^\top) + ||\mathbf{t}||_2^2 - 2\langle \mathbf{w}_j, \mathbf{t}\rangle - 2^\top(\mathbf{w}_j s_j^\top(\mathtt{RA})^\top) + 2\langle \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle \tag{A.12}$$

Consequently,

$$\sum_{j\in m} ||\mathbf{w}_j - \mathtt{RA}\mathbf{s}_j - \mathbf{t}||^2 = \sum_{j\in m} ||\mathbf{w}_j||^2 + tr(\mathtt{RA}(\sum_{j\in m} \mathbf{s}_j\mathbf{s}_j^\top)(\mathtt{RA})^\top) + \sum_{j\in m} ||\mathbf{t}||^2 \tag{A.13}$$

$$- 2\langle \sum_{j\in m} \mathbf{w}_j, \mathbf{t}\rangle - 2tr((\sum_{j\in m} \mathbf{w}_j\mathbf{s}_j^\top)(\mathtt{RA})^\top)$$

$$+ 2\langle \sum_{j\in m} \mathbf{s}_j, (\mathtt{RA})^\top \mathbf{t}\rangle. \tag{A.14}$$

This allows the cost function $C^{i,\alpha} = (\mathtt{A}_i, \mathtt{R}_i, \mathtt{T}_i)$ and its derivatives to be computed in constant time given the pre-computed values:

$$\sum_{j\in m}||\mathbf{w}_j||^2 \quad \sum_{j\in m}s_j s_j^\top, \quad \sum_{j\in m}1, \quad \sum_{j\in m}\mathbf{w}_j, \quad \sum_{j\in m}\mathbf{w}_j s_j^\top, \quad \text{and} \sum_{j\in m}s_j.$$

See Section 6.2.1 for discussion.